

VOICE CODE CONVERSION APPARATUS

Patent number: JP2002202799
 Publication date: 2002-07-19
 Inventor: SUZUKI MASANAO; OTA TAKASHI; TSUCHINAGA YOSHITERU
 Applicant: FUJITSU LTD
 Classification:
 - international: G10L19/04; G10L19/12; G10L19/00
 - european:
 Application number: JP20010075427 20010316
 Priority number(s):

Also published as:

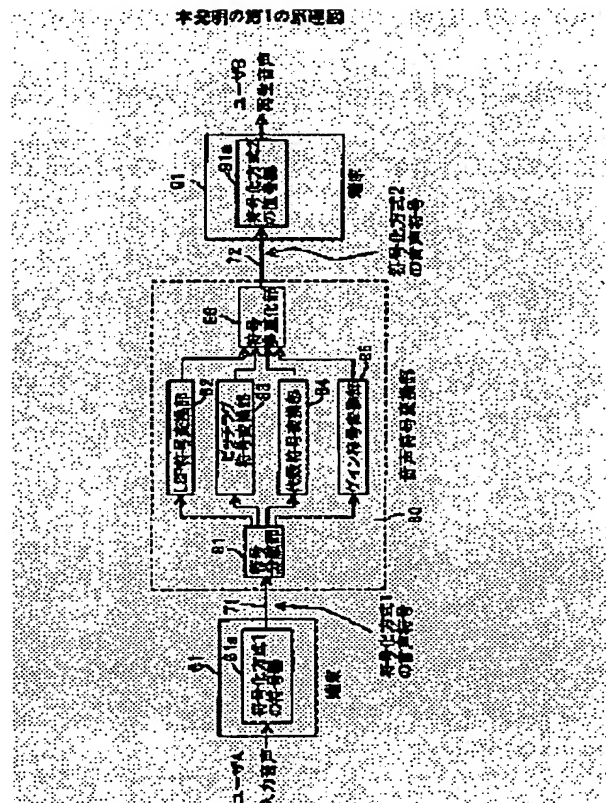
EP1202251 (A2)
 US2002077812 (A)
 EP1202251 (A3)

Report a data error

Abstract of JP2002202799

PROBLEM TO BE SOLVED: To prevent the degradation of the quality of reproduced voice and to reduce voice delays, even if a voice code is converted from a code of a first voice encoding system to that of a second voice-encoding system.

SOLUTION: A voice code converting apparatus 80 receives voice codes obtained by encoding with the first voice encoding system, and converts and outputs the voice codes into voice codes of the second voice-encoding system. In this voice code converting apparatus 80, a code separation means 81 separates a plurality of component codes required for reproduction of an audio signal from the voice codes of the first voice encoding system. Individual code converting parts 82 to 85 subjects the codes of individual components, respectively to inverse quantization, and subsequently quantize the dequantized value, by using the quantization table of the second voice-encoding system to generate codes. A code multiplexer 86 multiplexes the codes outputted from each quantization part and outputs the voice codes of the second voice encoding system.



Data supplied from the esp@cenet database - Patent Abstracts of Japan

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開2002-202799

(P2002-202799A)

(43) 公開日 平成14年7月19日 (2002.7.19)

(51) Int.Cl. ⁷	識別記号	F I	テーマコード(参考)
G 1 0 L 19/04		G 1 0 L 9/14	J 5 D 0 4 5
19/12			S
19/00		9/18	E

審査請求 未請求 請求項の数10 O L (全 35 頁)

(21) 出願番号 特願2001-75427(P2001-75427)

(22) 出願日 平成13年3月16日(2001.3.16)

(31) 優先権主張番号 特願2000-330108(P2000-330108)

(32) 優先日 平成12年10月30日(2000.10.30)

(33) 優先権主張国 日本 (J P)

(71) 出願人 000005223
富士通株式会社
神奈川県川崎市中原区上小田中4丁目1番
1号

(72) 発明者 鈴木 政直
神奈川県川崎市中原区上小田中4丁目1番
1号 富士通株式会社内

(72) 発明者 大田 恭士
神奈川県川崎市中原区上小田中4丁目1番
1号 富士通株式会社内

(74) 代理人 100084711
弁理士 斉藤 千幹

最終頁に続く

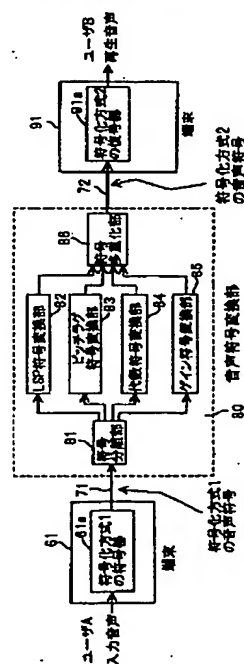
(54) 【発明の名称】 音声符号変換装置

(57) 【要約】

【課題】 音声符号を第1の音声符号化方式から第2の音声符号化方式に変換しても、再生音声の品質を劣化させず、しかも、音声の遅延を短縮する。

【解決手段】 第1の音声符号化方式により符号化して得られる音声符号を入力され、該音声符号を第2の音声符号化方式の音声符号に変換して出力する音声符号変換装置である。この音声符号変換装置80において、符号分離手段81は第1の音声符号化方式による音声符号より音声信号を再現するために必要な複数の成分の符号を分離し、各符号変換部82～85は各成分の符号をそれぞれ逆量子化し、ついで、逆量子化値を第2の音声符号化方式の量子化テーブルを用いて量子化して符号を発生し、符号多重部86は各量子化部から出力する符号を多重して第2の音声符号化方式による音声符号を出力する。

本発明の第1の原理図



【特許請求の範囲】

【請求項1】 第1の音声符号化方式により符号化して得られる音声符号を入力され、該音声符号を第2の音声符号化方式の音声符号に変換して出力する音声符号変換装置において、

第1の音声符号化方式による音声符号より、音声信号を再現するために必要な複数の成分の符号を分離する符号分離手段、

各成分の符号をそれぞれ逆量子化して逆量子化値を出力する逆量子化部、

前記各逆量子化部から出力する逆量子化値を第2の音声符号化方式により量子化して符号を発生する量子化部、各量子化部から出力する符号を多重して第2の音声符号化方式による音声符号を出力する手段、

を備えたことを特徴とする音声符号変換装置。

【請求項2】 音声信号の一定サンプル数をフレームとし、フレーム毎の線形予測分析により得られる線形予測係数(LPC係数)又は該LPC係数から求まるLSPパラメータを量子化することにより得られる第1のLPC符号と、周期性音源信号を出力するための適応符号帳の出力信号を特定する第1のピッチラグ符号と、雑音性音源信号を出力するための雑音符号帳の出力信号を特定する第1の雑音符号と、前記適応符号帳の出力信号の振幅を表す適応符号帳ゲインと前記雑音符号帳の出力信号の振幅を表す雑音符号帳ゲインとを量子化して得られる第1のゲイン符号とを求め、これら符号で音声信号を符号化する方式を第1の音声符号化方式とし、第1の音声符号化方式と異なる量子化方法により量子化して得られる第2のLPC符号、第2のピッチラグ符号、第2の雑音符号、第2のゲイン符号とで音声信号を符号化する方式を第2の音声符号化方式とするとき、

第1の音声符号化方式により符号化した音声符号を入力され、該音声符号を第2の音声符号化方式の音声符号に変換する音声符号変換装置において、

前記第1のLPC符号を第1の音声符号化方式のLPC逆量子化方法により逆量子化し、得られるLPC係数の逆量子化値を第2の音声符号化方式のLPC量子化テーブルを用いて量子化して第2のLPC符号を求めるLPC符号変換手段、第1の音声符号化方式におけるピッチラグ符号と第2の音声符号化方式におけるピッチラグ符号との相違を考慮した変換処理により、前記第1のピッチラグ符号を第2のピッチラグ符号に変換するピッチラグ変換手段、

第1の音声符号化方式における雑音符号と第2の音声符号化方式における雑音符号との相違を考慮した変換処理により、前記第1の雑音符号を第2の雑音符号に変換する雑音符号変換手段、

前記第1のゲイン符号を第1の音声符号化方式のゲイン逆量子化方法により逆量子化してゲイン逆量子化値を求めるゲイン逆量子化手段、

前記ゲイン逆量子化値を第2の音声符号化方式のゲイン

量子化テーブルを用いて量子化して第2のゲイン符号に変換するゲイン符号変換手段、

とを有することを特徴とする音声符号変換装置。

【請求項3】 前記ゲイン逆量子化手段は、前記第1のゲイン符号を第1の音声符号化方式のゲイン逆量子化方法により逆量子化して適応符号帳ゲインの逆量子化値と雑音符号帳ゲインの逆量子化値を求め前記ゲイン符号変換手段は、前記適応符号帳ゲイン及び雑音符号帳ゲインの逆量子化値をそれぞれ個別に第2の音声符号化方式のゲイン量子化テーブルを用いて量子化して適応符号帳ゲイン符号及び雑音符号帳ゲイン符号を発生し、これら2つのゲイン符号で前記第2のゲイン符号を構成する、ことを特徴とする請求項2記載の音声符号変換装置。

【請求項4】 前記ゲイン符号変換手段は、前記適応符号帳ゲインの逆量子化値を第2の音声符号化方式のゲイン量子化テーブルを用いて量子化して適応符号帳ゲイン符号を発生する第1ゲイン符号変換手段、前記雑音符号帳ゲインの逆量子化値を第2の音声符号化方式のゲイン量子化テーブルを用いて量子化して雑音符号帳ゲイン符号を発生する第2ゲイン符号変換手段、を有することを特徴とする請求項3記載の音声符号変換装置。

【請求項5】 音声信号の一定サンプル数をフレームとし、フレーム毎の線形予測分析により得られる線形予測係数(LPC係数)又は該LPC係数から求まるLSPパラメータを量子化することにより得られる第1のLPC符号と、周期性音源信号を出力するための適応符号帳の出力信号を特定する第1のピッチラグ符号と、雑音性音源信号を出力するための雑音符号帳の出力信号を特定する第1の雑音符号と、前記適応符号帳の出力信号の振幅を表す適応符号帳ゲインを量子化して得られる第1の適応符号帳ゲイン符号と、前記雑音符号帳の出力信号の振幅を表す雑音符号帳ゲインを量子化して得られる第1の雑音符号帳ゲイン符号とを求め、これら符号で音声信号を符号化する方式を第1の音声符号化方式とし、第1の音声符号化方式と異なる量子化方法により量子化して得られる第2のLPC符号、第2のピッチラグ符号、第2の雑音符号、第2のゲイン符号とで音声信号を符号化する方式を第2の音声符号化方式とするとき、

第1の音声符号化方式により符号化した音声符号を入力され、該音声符号を第2の音声符号化方式の音声符号に変換する音声符号変換装置において、

前記第1のLPC符号を第1の音声符号化方式のLPC逆量子化方法により逆量子化し、得られるLPC係数の逆量子化値を第2の音声符号化方式のLPC量子化テーブルを用いて量子化して第2のLPC符号を求めるLPC符号変換手段、第1の音声符号化方式におけるピッチラグ符号と第2の音声符号化方式におけるピッチラグ符号との相違を考慮した変換処理により、前記第1のピッチラグ符号を第2のピッチラグ符号に変換するピッチラグ変換手段、

第1の音声符号化方式における雑音符号と第2の音声符号化方式における雑音符号との相違を考慮した変換処理により、前記第1の雑音符号を第2の雑音符号に変換する雑音符号変換手段、

前記第1の適応符号帳ゲイン符号を第1の音声符号化方式のゲイン逆量子化方法により逆量子化して得られる逆量子化値と、前記第1の雑音符号帳ゲイン符号を第1の音声符号化方式のゲイン逆量子化方法により逆量子化して得られる逆量子化値をまとめて、第2の音声符号化方式のゲイン量子化テーブルを用いて量子化して前記第2のゲイン符号を発生するゲイン符号変換手段、を有することを特徴とする音声符号変換装置。

【請求項6】 前記LPC符号変換手段は、前記第1のLPC符号の逆量子化値と前記求めた第2のLPC符号の逆量子化値との間の第1の距離を演算する第1演算部、

現フレームの第2のLPC符号の逆量子化値と前フレームの第2のLPC符号の逆量子化値を用いて中間の第2のLPC符号の逆量子化値を補間する補間部、

中間の第1のLPC符号の逆量子化値と前記補間により求まる中間の第2のLPC符号の逆量子化値との間の第2の距離を演算する第2演算部、

第1、第2の距離の和が最小となるように、前記LPC係数の逆量子化値を第2のLPC符号に符号化する符号部、を備えたことを特徴とする請求項2又は請求項5記載の音声符号変換装置。

【請求項7】 第1、第2の距離に重み付けする重み付け手段を備え、

前記符号部は、重み付けされた第1、第2の距離の和が最小となるように、前記LPC係数の逆量子化値を第2のLPC符号に符号化する、

ことを特徴とする請求項6記載の音声符号変換装置。

【請求項8】 前記LPC符号変換手段は、LPC係数又はLSPパラメータを n 次のベクトルで表現し、 n 次のベクトルを複数の小ベクトルに分割した時、小ベクトル毎に前記第1、第2の距離の和が小さい複数の符号候補を算出する符号候補算出手段、

各小ベクトルの複数の符号候補の中から1つづつ符号を選んでLPC係数逆量子化値の n 次のLPC符号とするとき、前記第1、第2の距離の和が最小となる n 次のLPC符号を決定し、該LPC符号を前記第2の符号とするLPC符号決定手段、

を備えたことを特徴とする請求項6又は請求項7記載の音声符号変換装置。

【請求項9】 フレーム毎に音響信号を第1の符号化方式により符号化して得られる音響符号を入力され、該音響符号を第2の符号化方式の音響符号に変換して出力する音響符号変換装置において、

第1の音響符号化方式による音響符号より、音響信号を再現するために必要な複数の成分の符号を分離する符号

分離手段、

分離された複数の成分の各符号を第2の音響符号化方式の音響符号に変換する符号変換部、

伝送路誤りが発生していなければ分離された符号をそのまま符号変換部に入力し、伝送路誤りが発生していれば誤り隠蔽処理を施して得られる符号を符号変換部に入力する符号修正部各符号変換部から出力する符号を多重して第2の音響符号化方式による音響符号を出力する手段、

を備えたことを特徴とする音響符号変換装置。

【請求項10】 音響信号の一定サンプル数をフレームとし、フレーム毎の線形予測分析により得られる線形予測係数(LPC係数)又は該LPC係数から求まるLSPパラメータを量子化することにより得られる第1のLPC符号と、周期性音源信号を出力するための適応符号帳の出力信号を特定する第1のピッチラグ符号と、雑音性音源信号を出力するための雑音符号帳の出力信号を特定する第1の雑音符号と、前記適応符号帳の出力信号の振幅を表すピッチゲインと前記代数符号帳の出力信号の振幅を表す代数符号帳ゲインとを量子化して得られる第1のゲイン符号とを求め、これら符号で音響信号を符号化する方式を第1の音響符号化方式とし、第1の音響符号化方式と異なる量子化方法により量子化して得られる第2のLPC符号、第2のピッチラグ符号、第2の代数符号、第2のゲイン符号とで音響信号を符号化する方式を第2の音響符号化方式とするとき、第1の音響符号化方式により符号化した音響符号を入力され、該音響符号を第2の音響符号化方式の音響符号に変換する音響符号変換装置において、

第1の音響符号化方式による音響符号より、音響信号を再現するために必要な複数の成分の符号を分離する符号分離手段、

分離された複数の成分の各符号を第2の音響符号化方式の音響符号に変換する符号変換部、

伝送路誤りが発生していなければ分離された符号をそのまま符号変換部に入力し、伝送路誤りが発生していれば誤り隠蔽処理を施して得られる符号を符号変換部に入力する符号修正部各符号変換部から出力する符号を多重して第2の音響符号化方式による音響符号を出力する手段、

を備えたことを特徴とする音響符号変換装置。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は音声符号変換装置に係わり、特に、第1の音声符号化方式により符号化して得られる音声符号を入力され、該音声符号を第2の音声符号化方式の音声符号に変換して出力する音声符号変換装置に関する。

【0002】

【従来の技術】近年、携帯電話の加入者が爆発的に増加

しており、今後も利用者数が増加することが予想されている。また、インターネットを使った音声通信(Voice over IP; VoIP)は、企業内ネットワーク(イントラネット)や長距離電話サービスなどの分野で普及しつつある。携帯電話やVoIPなどの音声通信システムでは、通信回線を有効利用するために音声圧縮する音声符号化技術が用いられている。携帯電話では国によってあるいはシステムによって異なる音声符号化技術が用いられており、特に次世代の携帯電話システムとして期待されているW-CDMAでは、世界共通の音声符号化方式としてAMR(Adaptive Multi-Rate; 適応マルチレート)方式が採用されている。一方、VoIPでは音声符号化方式としてITU-T勧告G.729A方式が広く用いられている。AMR、G.729A方式は共にCELP(Code Excited Linear Prediction; 符号駆動線形予測符号化)と呼ばれる基本アルゴリズムを用いている。以下ではG.729A方式を例にしてCELPの動作原理を説明する。

【0003】・CELPの動作原理

CELPは、人間の声道特性を表す線形予測係数(LPC係数)、音声のピッチ成分と雑音成分とからなる音源信号を表わすパラメータを効率良く伝送することとを特徴とする。すなわち、CELPでは人間の声道を次式

【数1】

$$H(z) = \frac{1}{1 + \sum_{i=1}^P a_i z^{-i}} \quad (1)$$

で表されるLPC合成フィルタ $H(z)$ で近似し、 $H(z)$ への入力(音源信号)が、音声の周期性を表すピッチ周期成分と、ランダム性を表す雑音成分とに分離できると仮定する。CELPは、入力音声信号をそのまま復号器側へ伝送するのではなく、LPC合成フィルタのフィルタ係数及び励起信号のピッチ周期成分と雑音成分を抽出し、これらを量子化して得られる量子化インデックスを伝送することにより、高い情報圧縮を実現している。

【0004】・符号器の構成及び動作

図23はITU-T勧告G.729A方式の符号器の構成図である。図23において、1フレーム当り所定サンプル数(=N)の入力信号(音声信号)Xがフレーム単位でLPC分析部1に入力する。サンプリング速度を8kHz、1フレーム期間を10msecとすれば、1フレームは80サンプルである。LPC分析部1は、人間の声道を式(1)で表される全極型フィルタと見なし、このフィルタの係数 a_i ($i=1, \dots, p$)を求める。ここで、Pはフィルタ次数である。一般に、電話帯域音声の場合はpとして10~12の値が用いられる。LPC(線形予測)分析部1では、入力信号と先読み分の40サンプル及び過去の信号120サンプルの合計240サンプルを用いてLPC分析を行いLPC係数を求める。

【0005】パラメータ変換部2はLPC係数をLSP(線スペクトル対)パラメータに変換する。ここで、LSPパラメータは、LPC係数と相互に変換が可能な周波数領域のパラメータであり、量子化特性がLPC係数よりも優れていることから量子化はLSPの領域で行われる。LSP量子化部3は変換されたLSPパラメータを量子化してLSP符号とLSP逆量子化値を求める。LSP補間部4は、現フレームで求めたLSP逆量子化値と前フレームで求めたLSP逆量子化値によりLSP補間値を求める。すなわち、1フレームは5msの第1、第2の2つのサブフレームに分割され、LPC分析部1は第2サブフレームのLPC係数を決定するが、第1サブフレームのLPC係数は決定しない。そこで、LSP補間部4は、現フレームで求めたLSP逆量子化値と前フレームで求めたLSP逆量子化値を用いて補間演算により第1サブフレームのLSP逆量子化値を予測する。

【0006】パラメータ逆変換部5はLSP逆量子化値とLSP補間値をそれぞれLPC係数に変換してLPC合成フィルタ6に設定する。この場合、LPC合成フィルタ6のフィルタ係数として、フレームの第1サブフレームではLSP補間値から変換されたLPC係数が用いられ、第2サブフレームではLSP逆量子化値から変換したLPC係数が用いられる。尚、以降において1に添字のあるもの、例えば $lspi$, $li^{(n)}$, ...における1はアルファベットのエルである。LSPパラメータ $lspi$ ($i=1, \dots, p$)はLSP量子化部3でスカラー量子化やベクトル量子化などにより量子化された後、量子化インデックス(LSP符号)が復号器側へ伝送される。図24は量子化方法説明図であり、量子化テーブル3aにはインデックス番号1~nに対応させて多数の量子化LSPパラメータの組が記憶されている。距離演算部3bは次式

$$d = W \cdot \sum_i \{lsp_q(i) - lspi\}^2 \quad (i=1 \sim p)$$

により距離を演算する。そして、qを1~nまで変化させた時、最小距離インデックス検出部3cは距離dが最小となるqを求め、インデックスqをLSP符号として復号器側へ伝送する。Wは重み付け係数である。

【0007】次に音源とゲインの探索処理を行なう。音源とゲインはサブフレーム単位で処理を行う。CELPでは音源信号をピッチ周期成分と雑音成分の2つに分け、ピッチ周期成分の量子化には過去の音源信号系列を格納した適応符号帳7を用い、雑音成分の量子化には代数符号帳や雑音符号帳などを用いる。以下では、音源符号帳として適応符号帳7と代数符号帳8の2つを使用する典型的なCELP型の音声符号化方式について説明する。適応符号帳7は、インデックス1~Lに対応して順次1サンプル遅延したNサンプル分の音源信号(周期性信号という)を出力するようになっている。図25は1サブフレーム40サンプル(N=40)とした場合の適応符号帳7の構成図であり、最新の(L+39)サンプルのピッチ周期成分を記憶するバッファBFで構成され、インデックス1により1~40サンプルよりなる周期性信号が特定され、インデックス2により2~41サンプルよりなる周期性信号が特定され、...インデックスLによりL~L+39サンプルより

なる周期性信号が特定される。初期状態では適応符号帳7の中身は全ての振幅が0の信号が入っており、毎サブフレーム毎に時間的に一番古い信号をサブフレーム長だけ捨て、現サブフレームで求めた音源信号を適応符号帳7に格納するように動作する。

【0008】適応符号帳探索は、過去の音源信号を格納している適応符号帳7を用いて音源信号の周期性成分を同定する。すなわち、適応符号帳7から読み出す開始点を1サンプルづつ変えながら適応符号帳7内の過去の音源信号をサブフレーム長(=40サンプル)だけ取り出し、LPC合成フィルタ6に入力してピッチ合成信号 βAP_L を作成する。ただし、 P_L は適応符号帳7から取り出された遅れLに相当する過去の周期性信号(適応符号ベクトル)、AはLPC合成フィルタ6のインパルス応答、 β は適応符号帳ゲインである。

【0009】演算部9は入力音声Xと βAP_L の誤差電力 E_L を次式

$$E_L = |X - \beta AP_L|^2 \quad (2)$$

により求める。適応符号帳出力の重み付き合成出力を AP_L とし、 AP_L の自己相関を R_{pp} 、 AP_L と入力信号Xの相互相関を R_{xp} とすると、式(2)の誤差電力が最小となるピッチラグ L_{opt} における適応符号ベクトル P_L は、次式

【数2】

$$\begin{aligned} P_L &= \operatorname{argmax} \left(\frac{R_{xp}^2}{R_{pp}} \right) \\ &= \operatorname{argmax} \left(\frac{(X^T AP_L)^2}{(AP_L)^T (AP_L)} \right) \end{aligned} \quad (3)$$

により表される。すなわち、ピッチ合成信号 AP_L と入力信号Xとの相互相関 R_{xp} をピッチ合成信号の自己相関 R_{pp} で正規化した値が最も大きくなる読み出し開始点を最適な開始点とする。以上より、誤差電力評価部10は(3)式を満足するピッチラグ L_{opt} を求める。このとき、最適ピッチゲイン β_{opt} は次式

$$\beta_{opt} = R_{xp} / R_{pp} \quad (4)$$

で与えられる。

【0010】次に代数符号帳8を用いて音源信号に含まれる雑音成分を量子化する。代数符号帳8は、振幅が1又は-1の複数のパルスから構成される。例として、フレーム長が40サンプルの場合のパルス位置を図26に示す。代数符号帳8は、1フレームを構成するN(=40)サンプル点を複数のパルス系統グループ1~4に分割し、各パルス系統グループから1つのサンプル点を取り出してなる全組み合わせについて、各サンプル点で+1あるいは-1のパルスを有するパルス性信号を雑音成分として順次出力する。この例では、基本的に1フレームあたり4本のパルスが配置される。図27は各パルス系統グループ1~4に割り当てたサンプル点の説明図であり、

(1) パルス系統グループ1には8個のサンプル点0、5、10、15、20、25、30、35が割り当てられ、(2) パルス系統グループ2には8個のサンプル点1、6、11、16、21、26、31、36が割り当てられ、(3) パルス系統グループ3には8個のサンプル点2、7、12、17、22、27、32、37が割り当てられ、(4) パルス系統グループ4には16個のサンプル点3、4、8、9、13、14、18、19、23、24、28、29、33、34、38、39が割り当てられている。

【0011】パルス系統グループ1~3のサンプル点を表現するために3ビット、パルスの正負を表現するのに1 bit、トータル4 bitが必要であり、又、パルス系統グループ4のサンプル点を表現するために4 bit、パルスの正負を表現するのに1 bit、トータル5 bit必要である。従って、図26のパルス配置を有する雑音符号帳8から出力するパルス性信号を特定するために17bitが必要になり、パルス性信号の種類は 2^{17} ($=2^4 \times 2^4 \times 2^4 \times 2^5$)存在する。図26に示すように各パルス系統のパルス位置は限定されており、代数符号帳探索では各パルス系統のパルス位置の組み合わせの中から、再生領域で入力音声との誤差電力が最も小さくなるパルスの組み合わせを決定する。すなわち、適応符号帳探索で求めた最適ピッチゲイン β_{opt} とし、適応符号帳出力 P_L に該ゲイン β_{opt} を乗算して加算器11に入力する。これと同時に代数符号帳8より順次パルス性信号を加算器に11に入力し、加算器出力をLPC合成フィルタ6に入力して得られる再生信号と入力信号Xとの差が最小となるパルス性信号を特定する。具体的には、まず入力信号Xから適応符号帳探索で求めた最適な適応符号帳出力 P_L 、最適ピッチゲイン β_{opt} から次式により代数符号帳探索のためのターゲットベクトル X' を生成する。

$$【0012】X' = X - \beta_{opt} AP_L \quad (5)$$

この例では、パルスの位置と振幅(正負)を前述のように17bitで表現するため、その組み合わせは2の17乗通り存在する。ここで、k通り目の代数符号出力ベクトルを C_k とすると、代数符号帳探索では次式

$$D = |X' - G_C AC_k|^2 \quad (6)$$

の評価関数誤差電力Dを最小とする符号ベクトル C_k を求める。 G_C は代数符号帳ゲインである。式(6)を最小化することは、次式

【数3】

$$D' = \frac{(X'^T AC_k)^2}{(AC_k)^T (AC_k)} \quad (7)$$

を最大とする C_k 、すなわちkを探すことと等価である。以上より、誤差電力評価部10は代数符号帳の探索において、代数合成信号 AC_k とターゲット信号 X' の相互相関を代数合成信号 AC_k の自己相関で正規化した値が最も大きくなるパルス位置と極性の組み合わせを特定するkを探索する。

【0013】次にゲイン量子化について説明する。G.729A方式では代数符号帳ゲインは直接には量子化されず、適応符号帳ゲイン $G_a (= \beta_{opt})$ と代数符号帳ゲイン G_c の補正係数 γ をベクトル量子化する。ここで、代数符号帳ゲイン G_c と補正係数 γ との間には

$$G_c = g' \times \gamma$$

なる関係がある。 g' は過去の4サブフレームの対数利得から予測される現フレームの利得である。ゲイン量子化器12の図示しないゲイン量子化テーブル(ゲイン符号帳)には、適応符号帳ゲイン G_a と代数符号帳ゲインに対する補正係数 γ の組み合わせが128通り($= 2^7$)用意されている。ゲイン符号帳の探索方法は、①適応符号帳出力ベクトルと代数符号帳出力ベクトルに対して、ゲイン量子化テーブルの中から1組のテーブル値を取り出してゲイン可変部13、14に設定し、②ゲイン可変部13、14でそれぞれのベクトルにゲイン G_a 、 G_c を乗じてLPC合成フィルタ6に入力し、③誤差電力評価部10において入力信号 X との誤差電力が最も小さくなる組み合わせを選択する、ことにより行なう。

【0014】以上より、出力情報選択部15は、①LSPの量子化インデックスであるLSP符号、②ピッチラグ符号 L_{opt} 、(3)代数符号帳インデックスである代数符号、(4)ゲインの量子化インデックスであるゲイン符号を多重して回線データを作成し、復号器に伝送する。以上説明した通り、CELP方式は音声の生成過程をモデル化し、そのモデルの特徴パラメータを量子化して伝送することにより、音声を効率良く圧縮することができる。

【0015】・復号器の構成及び動作

図28にG.729A方式の復号器のブロック図である。符号器側から送られてきた回線データが回線復号部21へ入力されてLSP符号、ピッチラグ符号、代数符号、ゲイン符号が出力される。復号器ではこれらの符号に基づいて音声データを復号する。復号器の動作については、復号器の機能が符号器に含まれているため一部重複するが、以下で簡単に説明する。LSP逆量子化部22はLSP符号が入力すると逆量子化し、LSP逆量子化値を出力する。LSP補間部23は現フレームの第2サブフレームにおけるLSP逆量子化値と前フレームの第2サブフレームのLSP逆量子化値から現フレームの第1サブフレームのLSP逆量子化値を補間演算する。次に、パラメータ逆変換部24はLSP補間値とLSP逆量子化値をそれぞれLPC合成フィルタ係数へ変換する。G.729A方式のLPC合成フィルタ25は、最初の第1サブフレームではLSP補間値から変換されたLPC係数を用い、次の第2サブフレームではLSP逆量子化値から変換されたLPC係数を用いる。

【0016】適応符号帳26はピッチラグ符号が指示する読み出し開始位置からサブフレーム長($=40$ サンプル)のピッチ信号を出力し、雑音符号帳27は代数符号に対応する読み出し位置からパルス位置とパルスの極性を出力する。また、ゲイン逆量子化部28は入力されたゲ

イン符号より適応符号帳ゲイン逆量子化値と代数符号帳ゲイン逆量子化値を算出してゲイン可変部29、30に設定する。加算部31は適応符号帳出力に適応符号帳ゲイン逆量子化値を乗じて得られる信号と、代数符号帳出力に代数符号帳ゲイン逆量子化値を乗じて得られる信号とを加え合わせて音源信号を作成し、この音源信号をLPC合成フィルタ25に入力する。これにより、LPC合成フィルタ25から再生音声を得ることができる。尚、初期状態では復号器側の適応符号帳26の内容は全て振幅0の信号が入っており、サブフレーム毎に時間的に一番古い信号をサブフレーム帳だけ捨て、一方、現サブフレームで求めた音源信号を適応符号帳26に格納するように動作する。つまり、符号器と復号器の適応符号帳26は常に最新の同じ状態になるように維持される。

【0017】・G729A方式とAMR方式における符号化方法の相違

次に、G729A音声符号化方式とAMR音声符号化方式の違いについて説明する。図29はG.729A方式とAMRの主要諸元を比較した結果である。なお、AMRの符号化モードは全部で8種類あるが図29の諸元は全ての符号化モードで共通である。G729A方式とAMR方式は、入力信号の標準化周波数($=8\text{KHz}$)、サブフレーム長($=5\text{msec}$)、線形予測次数($=10$ 次)は同じであるが、図30に示すようにフレーム長が異なり、1フレーム当りのサブフレーム数が異なっている。G.729A方式では1フレームは2つの第0～第1サブフレームで構成され、AMR方式では1フレームは4つの第0～第3サブフレームで構成されている。

【0018】図31はG.729A方式とAMR方式におけるビット割り当ての比較結果を示すもので、AMR方式についてはG.729Aのビットレートに最も近い7.95kbit/sモードの場合を示した。図31から明らかなように、1サブフレーム当りの代数符号帳のビット数($=17$ ビット)は同じであるが、その他の符号に必要なビット数の配分は全て異なっている。また、G.729A方式では適応符号帳ゲインと代数符号帳ゲインをまとめてベクトル量子化するため、ゲイン符号は1サブフレームにつき1種類であるが、AMR方式では1サブフレームにつき適応符号帳ゲインと代数符号帳ゲインの2種類が必要である。以上説明した通り、インターネットで音声を通信するVoIPで広く用いられているG.729A方式と次世代携帯電話システムで採用されたAMR方式とでは、基本アルゴリズムが共通であるが、フレーム長が異なり、しかも、符号を表現するビット数が異なっている。

【0019】

【発明が解決しようとする課題】インターネットと携帯電話の普及に伴い、インターネットユーザと携帯電話網のユーザによる音声通話の通信量が今後ますます増えてくると考えられる。図29はかかる場合におけるネットワークとユーザの関係を示す概念図である。ネットワーク(例えばインターネット)51のユーザAが、ネット

ワーク（例えば携帯電話網）53のユーザBと音声通信をする場合、ネットワーク51の音声通信で用いられる第1の符号化方式とネットワーク53の音声通信で用いられる第2の符号化方式が異なる場合、ユーザ間で通信をすることはできない。そこで、図32に示すように音声符号変換部55をネットワーク間に設け、音声符号変換部55において一方のネットワークで符号化された音声符号を他方のネットワークで用いられる符号化方式の音声符号に変換する。

【0020】図33は音声符号変換を用いた従来技術の例である。ユーザAが端末52に対して入力した音声ユーザBの端末54に伝える場合のみを考える。ここで、ユーザAの持つ端末52は符号化方式1の符号器52aのみを持ち、ユーザBの持つ端末54は符号化方式2の復号器54aのみを持つこととする。送信側のユーザAが発した音声は、端末52に組み込まれた符号化方式1の符号器52aへ入力する。符号器52aは入力した音声信号を符号化方式1の音声符号に符号化して伝送路51'に送出する。音声符号変換部55の復号器55aは、伝送路51'を介して符号化方式1の音声符号が入力すると、符号化方式1の音声符号から一旦再生音声を復号する。続いて、音声符号変換部55の符号器55bは再生音声信号を符号化方式2の音声符号に変換して伝送路53'に送出する。この符号化方式2の音声符号は伝送路53'を通して端末54に入力する。復号器54aは符号化方式2の音声符号が入力すると、符号化方式2の音声符号から再生音声を復号する。これにより、受信側のユーザBは再生音声を聞くことができる。以上のように一度符号化された音声を復号し、復号された音声を再度符号化する処理をタンデム接続と呼ぶ。

【0021】一度符号化処理により情報圧縮された音声（再生音声）は、元の音声（原音）に比べて音声の情報量が減っており、再生音声の音質は原音よりも悪い。特に、G.729A方式やAMR方式に代表される近年の低ビットレート音声符号化方式では、高圧縮率を実現するために入力音声に含まれる多くの情報を捨てて符号化しており、符号化と復号を繰り返すタンデム接続を行うと、再生音声の品質が著しく劣化するという問題があった。また、タンデム処理には遅延の問題がある。電話のような双方向通信では、100ミリ秒以上の遅延があると該遅延が話者に認識され、会話に支障をきたすことが知られている。フレーム処理を行う音声符号化方式において実時間処理ができたとしても基本的にフレーム長の4倍の遅延が避けられないことが知られている。例えば、AMR方式のフレーム長は20ミリ秒であるから、遅延は最低でも80ミリ秒となる。従来の音声符号変換方法では、G.729A方式とAMR方式のタンデム接続が必要となり、その際の遅延は160ミリ秒以上になってしまい、通話の際に遅延が話者に認識され、会話に支障をきたすという問題があった。

【0022】以上説明した通り、異なる音声符号化方式を採用するネットワーク間で音声通信を行うため、従来技術は圧縮された音声符号を一旦音声に復号し、再び音声符号化をするタンデム処理を行うため、再生音声の品質が著しく劣化するという問題と、遅延により会話に支障をきたすという問題があった。また、従来技術は伝送路誤りの影響を考慮していない問題がある。すなわち、携帯電話のように無線通信を用いる場合、フェージング等の影響によりビット誤りやバースト誤りが発生し、音声符号が本来と異なるものに变化したり、1フレーム全部の音声符号が欠落してしまう場合がある。また、インターネットでは網が混雑していると伝送遅延が大きくなり、1フレーム全部の音声符号が欠落したり、フレームの順番が入れ替わってしまう場合がある。このように伝送路誤りが混入すると誤った音声符号を基に変換が行われるため、最適な音声符号に変換できなくなる。このため、伝送路誤りによる影響を軽減する技術が求められている。以上より、本発明の目的は音声符号を第1の音声符号化方式から第2の音声符号化方式に変換しても、再生音声の品質が劣化しないようにすることである。本発明の別の目的は、音声符号を第1の音声符号化方式から第2の音声符号化方式に変換しても、音声の遅延を小さくでき、良好な会話を行なえるようにすることである。本発明の別の目的は、伝送路誤りによって変形された音声符号から誤りの影響をできるだけ軽減し、誤りの影響が軽減された音声符号を音声符号変換することにより、伝送路誤りによる再生音声の音質劣化を小さくすることである。

【0023】

【課題を解決するための手段】本発明の音声符号変換装置は、第1の音声符号化方式により符号化して得られる音声符号を入力され、該音声符号を第2の音声符号化方式の音声符号に変換して出力するものであり、(1) 第1の音声符号化方式による音声符号より、音声信号を再現するために必要な複数の成分の符号を分離する符号分離手段、(2) 各成分の符号をそれぞれ逆量子化して逆量子化値を出力する逆量子化部、(3) 前記各逆量子化部から出力する逆量子化値を第2の音声符号化方式の量子化テーブルを用いて量子化して符号を発生する量子化部、(4) 各量子化部から出力する符号を多重して第2の音声符号化方式による音声符号を出力する手段、を備えている。この音声符号変換装置によれば、第1の音声符号化方式による音声符号を逆量子化し、逆量子化値を第2の音声符号化方式の量子化テーブルを用いて量子化して符号化するため、音声符号変換の過程で再生音声を出力する必要がない。このため、最終的に再生される音声の品質劣化を抑えることができ、しかも、処理時間を短縮して信号遅延を小さくすることができる。本発明の他の音声符号変換装置は、フレーム毎に音響信号を第1の符号化方式により符号化して得られる音響符号を入力され、

該音響符号を第2の符号化方式の音響符号に変換して出力するものであり、(1) 第1の符号化方式による音響符号より、音響信号を再現するために必要な複数の成分の符号を分離する符号分離手段、(2) 伝送路誤りが発生していなければ前記各成分の符号をそれぞれ逆量子化して逆量子化値を出力すると共に、伝送路誤りが発生していれば誤り隠蔽処理を施して得られる逆量子化値を出力する逆量子化部、(3) 前記各逆量子化部から出力する逆量子化値を第2の符号化方式の量子化テーブルを用いて量子化して符号を発生する量子化部、(4) 各量子化部から出力する符号を多重して第2の符号化方式による音響符号を出力する手段、を備えている。この音声符号変換装置によれば、伝送路誤りが発生していれば誤り隠蔽処理を施して得られる逆量子化値を量子化して別の符号を発生するため、伝送路誤りによる再生音声の音質劣化を軽減することができる。

【0024】

【発明の実施の形態】(A) 本発明の原理

図1は本発明の音声符号変換装置の原理図である。音声符号変換装置は第1の音声符号化方式(符号化方式1)により符号化した音声符号を入力され、該音声符号を第2の音声符号化方式(符号化方式2)の音声符号に変換して出力する。端末61に組み込まれた符号化方式1の符号器61aはユーザAが発した音声信号を符号化方式1の音声符号に符号化して伝送路71に送出する。音声符号変換部80は伝送路71より入力した符号化方式1の音声符号を符号化方式2の音声符号に変換して伝送路72に送出し、端末91の復号器91aは、伝送路72を介して入力する符号化方式2の音声符号から再生音声を復号し、ユーザBはこの再生音声を聞くことができる。

【0025】符号化方式1は、①フレーム毎の線形予測分析により得られる線形予測係数(LPC計数)又は該LPC係数から求まるLSPパラメータを量子化することにより得られる第1のLPC符号と、②周期性音源信号を出力するための適応符号帳の出力信号を特定する第1のピッチラグ符号と、③雑音性音源信号を出力するための雑音符号帳の出力信号を特定する第1の雑音符号と、④前記適応符号帳の出力信号の振幅を表す適応符号帳ゲインと前記雑音符号帳の出力信号の振幅を表す雑音符号帳ゲインとを量子化して得られる第1のゲイン符号とで音声信号を符号化する方式である。又、符号化方式2は、第1の音声符号化方式と異なる量子化方法により量子化して得られる①第2のLPC符号、②第2のピッチラグ符号、③第2の雑音符号、④第2のゲイン符号とで音声信号を符号化する方式である。

【0026】音声符号化部80は、符号分離部81、LSP符号変換部82、ピッチラグ符号変換部83、代数符号変換部84、ゲイン符号変換部85、符号多重化部86を有している。符号分離部81は、端末61の符号器61aから伝送路71を介して入力する符号化方式1に

よる音声符号より、音声信号を再現するために必要な複数の成分の符号、すなわち、①LSP符号、②ピッチラグ符号、③代数符号、④ゲイン符号に分離し、それぞれを各符号変換部82～85に入力する。各符号変換部82～85は入力された音声符号化方式1によるLSP符号、ピッチラグ符号、代数符号、ゲイン符号をそれぞれ音声符号化方式2によるLSP符号、ピッチラグ符号、代数符号、ゲイン符号に変換し、符号多重化部86は変換された音声符号化方式2の各符号を多重化して伝送路72に送出する。

【0027】図2は各符号変換部82～85の構成を明示した音声符号変換部の構成図であり、図1と同一部分には同一符号を付している。回線データ分離部81は伝送路より入力端子#1を介してに入力する回線データ(符号化方式1による音声符号)より、LSP符号1、ピッチラグ符号1、代数符号1、ゲイン符号1を分離し、それぞれ符号変換部82～85に入力する。LSP符号変換部82のLSP逆量子化器82aは、符号化方式1のLSP符号1を逆量子化してLSP逆量子化値を出力し、LSP量子化器82bは該LSP逆量子化値を符号化方式2のLSP量子化テーブルを用いて量子化してLSP符号2を出力する。ピッチラグ符号変換部83のピッチラグ逆量子化器83aは、符号化方式1のピッチラグ符号1を逆量子化してピッチラグ逆量子化値を出力し、ピッチラグ量子化器83bは該ピッチラグ逆量子化値を符号化方式2のピッチラグ量子化テーブルを用いて量子化してピッチラグ符号2を出力する。代数符号変換部84の代数符号逆量子化器84aは、符号化方式1の代数符号1を逆量子化して代数符号逆量子化値を出力し、代数符号量子化器84bは該代数符号逆量子化値を符号化方式2の代数符号量子化テーブルを用いて量子化して代数符号2を出力する。ゲイン符号変換部85のゲイン逆量子化器85aは、符号化方式1のゲイン符号1を逆量子化してゲイン逆量子化値を出力し、ゲイン量子化器85bは該ゲイン逆量子化値を符号化方式2のゲイン量子化テーブルを用いて量子化してゲイン符号2を出力する。

【0028】回線データ多重化部86は、各量子化器82b～85bから出力するLSP符号2、ピッチラグ符号2、代数符号2、ゲイン符号2を多重して回線データ(符号化方式2による音声符号)を作成して出力端子#2より伝送路に送出する。従来は、符号化方式1で符号化された音声符号を一旦音声に復号して得られた再生音声を入力とし、再度符号化と復号を行っていた。このため、再度の符号化(つまり音声情報圧縮)によって原音に比べて遥かに情報量が少なくなっている再生音声から音声のパラメータ抽出を行うため、それによって得られる音声符号は必ずしも最適なものではなかった。これに対し、本発明の音声符号化装置によれば、符号化方式1の音声符号を逆量子化及び量子化の過程を介して符号化方式2の音声符号に変換するため、従来のタンデム接続に

比べて格段に劣化の少ない音声符号変換が可能となる。また、音声符号変換のために一度も音声に復号する必要がないので、従来のタンデム接続で問題となっていた遅延も少なく済むという利点がある。

【0029】(B) 第1実施例

図3は本発明の第1実施例の音声符号変換部の構成図であり、図2の原理図と同一部分には同一符号を付している。図2の原理図と異なる点は、バッファ87を設けた点、及び、ゲイン符号変換部85のゲイン量子化器を適応符号帳ゲイン量子化器85b₁と雑音符号帳ゲイン量子化器85b₂で構成している点である。又、図3の第1実施例において、符号化方式1としてG.729A符号化方式を用い、符号化方式2としてAMR符号化方式を用いるものとする。又、AMR符号化方式には8つの符号化モードが存在するが、第1実施例では伝送レート7.95kbit/sの符号化モードを用いるものとする。

【0030】図3において、G.729A方式の符号器（図示せず）から伝送路を介して第nフレーム目の回線データbst1(n)が端子#1に入力する。ここでG.729A符号化方式のビットレートは8kbit/sであるから、回線データbst1(n)は80ビットのビット系列で表される。回線データ分離部81は、回線データbst1(n)からLSP符号I_LSP1(n)、ピッチラグ符号I_LAG1(n, j)、代数符号I_CODE1(n, j)、ゲイン符号I_GAIN1(n, j)を分離して各変換部82～85に入力する。ここで、添字jはフレームを構成する第0、第1のサブフレームの番号を表し、0、1の値をとる。

【0031】(a) LSP符号変換部

まず、LSP符号変換部82の動作について説明する。図4はG.729A符号化方式とAMR符号化方式におけるフレームとLSP量子化の関係を示す。図4(a)に示すようにG.729A方式のフレーム長は10msecであり、10msecに1回だけ第1サブフレームの入力音声信号から求めたLSPパラメータを量子化する。これに対し、AMR方式のフレーム長は20msecであり、20msecに1回だけ第3サブフレームの入力信号からLSPパラメータを量子化する。つまり、同じ20msecを単位として考えると、G.729A方式は2回のLSP量子化を行うのに対してAMR方式は1回しか量子

化を行わない。このため、G.729A方式の連続する2つのフレームのLSP符号そのままではAMR方式のLSP符号に変換することはできない。

【0032】そこで、第1実施例では、奇数フレームのLSP符号のみをAMR方式のLSP符号に変換し、偶数フレームのLSP符号は変換しない構成とした。ただし、偶数フレームのLSP符号をAMR方式のLSP符号に変換して、奇数フレームのLSP符号を変換しないようにすることもできる。又、G.729A方式のLSP逆量子化器82aは以下で説明するようにフレーム間予測を用いるので、状態更新は毎フレーム行われる。LSP逆量子化器82aは奇数フレームのLSP符号I_LSP1(n+1)が入力すると該符号を逆量子化してLSP逆量子化値lsp(i), (i=1, ..., 10)を出力する。ここで、LSP逆量子化器82aはG.729A符号化方式の復号器において用いられる逆量子化器と同じ動作をする。次に、LSP量子化器82bはLSP逆量子化値lsp(i)が入力するとAMR符号化方式に従って量子化してLSP符号I_LSP2(n)を求める。ここで、LSP量子化器82bはAMR方式の符号器において用いられる量子化器と必ずしもまったく同じものである必要はないが、少なくともLSP量子化テーブルはAMR方式の量子化テーブルと同一のテーブルを用いるものとする。

【0033】LSP逆量子化器におけるLSP逆量子化方法LSP逆量子化器82aにおけるG.729A方式のLSP逆量子化方法をG.729に沿って説明する。第nフレームのLSP符号I_LSP1(n)が入力すると、LSP逆量子化器82aはLSP符号I_LSP1(n)を4つの符号L₀, L₁, L₂, L₃に分割する。ここで、符号L₁は第1のLSP符号帳CB1の要素番号（インデックス番号）を表し、符号L₂, L₃はそれぞれ第2、第3のLSP符号帳CB2、CB3の要素番号を表す。第1のLSP符号帳CB1は10次元のベクトルを128組持ち、第2、第3のLSP符号帳CB2、CB3は共に5次元ベクトルを32組持つ。符号L₀は後述する2種類のMA予測係数のうちどちらを使うかを表す。

【0034】ついで、第nフレーム目の残差ベクトルl_i(n)を次式

【数4】

$$l_i^{(n)} = \begin{cases} CB1(L_1, i) + CB2(L_2, i) & (i = 1, \dots, 5) \\ CB1(L_1, i) + CB3(L_3, i - 5) & (i = 6, \dots, 10) \end{cases} \quad (8)$$

により求める。同様にして第(n+1)フレーム目の残差ベクトルl_i⁽ⁿ⁺¹⁾を求めることができる。第(n+1)フレーム目の残差ベクトルl_i⁽ⁿ⁺¹⁾と過去4フレームで求めた残

差ベクトルl_i^(n+1-k)から次式

【数5】

$$\omega(i) = (1 - \sum_{k=1}^4 p(i, k)) l_i^{(n+1)} + \sum_{k=1}^4 p(i, k) l_i^{(n+1-k)}, \quad (i = 1, \dots, 10) \quad (9)$$

によりLSF係数ω(i)を求める。p(i, k)は2種類のMA予測係数のうち符号L₀により指定された方の係数を表す。なお、第nフレーム目については、残差ベクトルからLS

F係数を求めていないが、その理由は第nフレームはLSP量子化部で量子化されないためである。ただし、残差ベクトルl_i⁽ⁿ⁾は状態更新のために必要である。ついで、L

SP逆量子化器82aは次式

$$lsp(i) = \cos(\omega(i)) \quad (i=1, \dots, 10) \quad (10)$$

を用いてLSF係数 $\omega(i)$ からLSP逆量子化値 $lsp(i)$ を求める。

【0035】・LSP量子化器におけるLSP量子化方法

LSP量子化器82bにおけるLSP量子化方法の詳細を説明する。AMR符号化方式では12.2kbit/sモードを除く他の7つのモードでは共通のLSP量子化方法を用いており、LSP符号帳のサイズのみが異なる。ここでは7.95kbit/sモードにおけるLSP量子化方法を説明する。(10)式によりLSP逆量子化値 $lsp(i)$ が求めればLSP量子化器82bは次式

$$r(i)^{(a)} = lsp(i) - q(i)^{(a)} \quad (11)$$

により、LSP逆量子化値 $lsp(i)$ から予測ベクトル $q(i)^{(a)}$ を差し引いて残差ベクトル $r(i)^{(a)}$ を求める。ここで、 m は現フレーム番号を表す。予測ベクトル $q(i)^{(a)}$ は1フレーム前の量子化された残差ベクトル $r^{(i)}(i)^{(a-1)}$ とMA予測係数 $a(i)$ を用いて次式

【数6】

$$q(i)^{(m)} = a(i)r^{(i)}(i)^{(m-1)} \quad (12)$$

により求まる。AMR符号化方式では10次元の残差ベクトル $r(i)^{(a)}$ をそれぞれ $r_1(i)$ ($i=1,2,3$)、 $r_2(i)$ ($i=4,5,6$)、 $r_3(i)$ ($i=7,8,9,10$)の3つの小ベクトルに分割し、それぞれを9ビットでベクトル量子化する。

【0036】ベクトル量子化はいわゆるパターンマッチング処理であり、LSP量子化器82bは予め用意された符号帳(各小ベクトルと同じ次元長の符号帳)CB1~CB3の中から、各小ベクトルとの重み付きユークリッド距離が最小となる符号帳ベクトルを最適な符号ベクトルとして選択する。そして、該最適な符号帳ベクトルが各符号帳CB1~CB3の何番目の要素であるかを表す番号(インデックス)を I_1, I_2, I_3 とすれば、これらインデックス I_1, I_2, I_3 を合成してなるLSP符号 $I_LSP2(m)$ を出力する。各符号帳CB1~CB3のサイズは全て9ビット(512組)であるから各インデックス I_1, I_2, I_3 の語長も9ビットとなり、LSP符号 $I_LSP2(m)$ は総計27ビットの語長を持つ。

【0037】図5はLSP量子化器82bの構成図であり、残差ベクトル算出部82b₁は(11)式により10次元の残差ベクトル

$$r(i) = r_1(i) \quad (i=1,2,3), r_2(i) \quad (i=4,5,6), r_3(i) \quad (i=7,8,9,10)$$

を出力する。最適符号帳ベクトル決定部82b₂~82b₄は、それぞれ小ベクトル $r_1(i)$ ($i=1,2,3$)、 $r_2(i)$ ($i=4,5,6$)、 $r_3(i)$ ($i=7,8,9,10$)との重み付きユークリッド距離が最小となる最適符号帳ベクトルのインデックス番号 I_1, I_2, I_3 を出力する。最適符号帳ベクトル決定部82b₂の低域用のLSP符号帳CB1にはインデックス1~512に対応させて512組の3次元の低域用のLSPベクトル r

($j,1$)、 $r(j,2)$ 、 $r(j,3)$ ($j=1\sim512$)が記憶されている。距離演算部DSCは次式

$$d = \sum_i \{r(j,i) - r_1(i)\}^2 \quad (i=1\sim3)$$

により距離を演算する。そして、 j を1~512まで変化させた時、最小距離インデックス検出部MDIは距離 d が最小となる j を求め、インデックス j を低域用のLSP符号 I_1 として出力する。最適符号帳ベクトル決定部82b₃~82b₄は図示しないが、中域用のLSP符号帳CB2、高域用のLSP符号帳CB3を用いて最適符号帳ベクトル決定部82b₂と同様にインデックス I_2, I_3 を出力する。

【0038】(b) ピッチラグ符号変換部

次に、ピッチラグ符号変換部83について説明する。前述した通り(図29参照)、G.729A符号化方式はフレーム長が10msecであるのに対して、AMR符号化方式のフレーム長は20msecである。このため、ピッチラグ符号を変換するにはG.729A方式の2フレームのピッチラグ符号をAMR方式の1フレーム分のピッチラグ符号として変換する必要がある。G.729A方式の第 n フレーム目と第 $(n+1)$ フレーム目のピッチラグ符号をAMR方式の第 m フレーム目のピッチラグ符号に変換する場合を考える。ここで、G.729A方式の第 n フレームの先頭フレームとAMR方式の第 m フレームの先頭フレームの時刻は等しいものとする。G.729A方式とAMR方式のフレーム、サブフレームの関係は図6(a)に示すようになる。又、G.729A方式とAMR方式の各サブフレームにおけるピッチラグの量子化ビット数は図6(b)に示すようになる(図31参照)。

【0039】以上より、偶数サブフレームではG.729A方式とAMR方式におけるピッチラグ符号の合成方法は全く同じであり、その量子化ビット数も同じ8ビットである。このことから偶数サブフレームについては、G.729A方式のピッチラグ符号をAMR方式のピッチラグ符号に次式により変換することができる。

$$I_LAG2(m,0) = I_LAG1(n,0) \quad (13)$$

$$I_LAG2(m,2) = I_LAG1(n+1,0) \quad (14)$$

また、奇数サブフレームについては、前サブフレームの整数ラグとの差分子量子化を行う点で共通であるが、量子化ビット数がAMR方式の方が1ビット多いことから、次式のように変換することができる。

【0040】

$$I_LAG2(m,1) = I_LAG1(n,1) + 15 \quad (15)$$

$$I_LAG2(m,3) = I_LAG1(n+1,1) + 15 \quad (16)$$

ここで、式(13)、(14)及び式(15)、(16)について詳しく説明する。G.729A方式及びAMR方式では、音声のピッチ周期を2.5msから18msecの間にあると仮定してピッチラグを決定する。ピッチラグを整数とすると符号化の処理が簡単であるが、ピッチ周期が短い場合には、周波数分解能が不足して音質が劣化してしまう。このためG.729A方式及びAMR方式では、サンプル補間フィルタを使用して1/3サンプル精度でピッチラグを決定している。すなわ

ち、適応符号帳にはあたかも実際のサンプリング周期の1/3の周期でサンプリングされた音声信号が記憶されたかようになる。以上より、ピッチラグには実際のサンプリング周期を示す整数ラグと1/3サンプリング周期を示す非整数ラグの2種類が存在する。

【0041】図7はG.729A方式におけるピッチラグとインデックスの関係を示すもので、図7(a)は偶数サブフレームの場合を示し、図7(b)は奇数サブフレームの場合を示している。偶数サブフレームでは、ラグの値が $19+1/3 \sim 85$ の範囲において1/3サンプル精度でインデックスを割り当て、 $85 \sim 143$ の範囲では1サンプルの精度でインデックスを割り当てている。ここで、ラグの整数部分を整数ラグと呼び、非整数部分(分数部分)を非整数ラグと呼ぶ。G.729A方式では偶数サブフレームのピッチラグに8ビット割り当てているので、ピッチラグインデックスは256通りである。例えば、ラグが $20+2/3$ の場合のインデックスは4となり、ラグが142の場合のインデックスは254となる。

【0042】一方、G.729A方式の奇数サブフレームでは、前サブフレーム(偶数サブフレーム)の整数ラグToldと現サブフレームのピッチラグ(整数ラグ、非整数ラグ)の差分を5ビット(32パターン)で量子化する。図7(b)に示すように、奇数サブフレームでは、Toldを基点とし、Toldのインデックスを17とする。Toldよりも $5+2/3$ サンプル小さいラグのインデックスを0とし、Toldよりも $4+2/3$ サンプル大きいラグのインデックスを31とする。つまり、 $Told-(5+2/3) \sim Told+(4+2/3)$ の範囲を1/3サンプル間隔で等分して32パターン(5ビット)のインデックスを割り当てている。

【0043】次に、AMR方式のピッチラグとインデックスの関係について説明する。図8はAMR方式のピッチラグとインデックスの関係説明図である。図8(a)は偶数サブフレームの場合を示し、図8(b)は奇数サブフレームの場合を示す。AMR方式の偶数サブフレームではピッチラグのインデックスに8ビットを割り当てる。ピッチラグは整数ラグと非整数ラグとから構成されており、インデックス番号の割り当て方法はG.729A方式と全く同じである。したがって、偶数サブフレームでは、G.729A方式のピッチラグインデックスを式(13)、(14)によりAMR方式のピッチラグインデックスに変換することができる。一方、AMR方式の奇数サブフレームでは、G.729A方式と同様に前サブフレームの整数ラグToldと現サブフレームのピッチラグの差分を量子化するが、量子化ビット数がG.729A方式よりも1ビット多く6ビット(64パターン)で量子化する。図8(b)に示すように、奇数サブフレームでは、Toldを基点とし、Toldのインデックスを32とする。Toldよりも $10+2/3$ サンプル小さいラグのインデックスを0とし、Toldよりも $9+2/3$ サンプル大きいラグのインデックスを63とする。つまり、 $Told-(10+2/3) \sim Told+(9+2/3)$ の範囲を1/3サンプル間隔で

等分して64パターン(6ビット)のインデックスを割り当てている。

【0044】図9は奇数サブフレームにおけるG.729A方式のインデックスをAMR方式のインデックスに変換する場合の対応関係図である。この図から分かるように同じラグの値であってもG.729A方式とAMR方式とではインデックスが全体的に15ずれている。例えば、ラグの $-(5+2/3)$ はG.729A方式では0番目のインデックスが割り当てられているが、AMR方式では15番目のインデックスが割り当てられている。したがって、奇数サブフレームにおけるG.729A方式のインデックスをAMR方式のインデックスに変換するには、(15)、(16)式に示すようにインデックスの値を15だけ加算して補正する必要がある。図3ではピッチラグ符号変換部83をピッチラグ逆量子化器83aとピッチラグ量子化器83bとで構成したが、ピッチラグ符号が量子化されていない場合には、ピッチラグ符号変換部83を(13)～(16)式の変換を行う変換ユニットで構成することができる。

【0045】(c) 代数符号変換

次に、代数符号の変換について説明する。G.729A方式とAMR方式とではフレーム長が異なるが、サブフレーム長は5ミリ秒(40サンプル)で共通である。すなわち、G.729A方式とAMR方式のフレーム、サブフレームの関係は図6(a)に示すようになる。又、G.729A方式とAMR方式の各サブフレームにおける代数符号の量子化ビット数は図6(c)に示すようになる(図28参照)。更に、両方式の代数符号帳は図26に示す構造を有しており、全く同じ構造となっている。したがって、G.729A方式の代数符号帳探索の出力結果である4本のパルス位置とパルスの極性情報は、そのままAMR方式の代数符号帳出力結果と一対一で置き換えることが可能であり、代数符号の変換式は次式となる。

$$I_CODE2(m,0)=I_CODE1(n,0) \quad (17)$$

$$I_CODE2(m,1)=I_CODE1(n,1) \quad (18)$$

$$I_CODE2(m,2)=I_CODE1(n+1,0) \quad (19)$$

$$I_CODE2(m,3)=I_CODE1(n+1,1) \quad (20)$$

図3では代数符号変換部83を代数符号逆量子化器84aと代数符号量子化器84bとで構成したが、代数符号が量子化されていない場合には、代数符号変換部84を(17)～(20)式の変換を行う変換ユニットで構成することができる。

【0046】(d) ゲイン符号変換

次にゲイン符号の変換について説明する。まず、ゲイン符号 $I_GAIN(n,0)$ をゲイン逆量子化器85a(図3)に入力する。G.729A方式ではゲインの量子化にベクトル量子化を用いているので、ゲイン逆量子化値としては適応符号帳ゲイン逆量子化値 G_a と、代数符号帳ゲインに対する補正係数の逆量子化値 γ_c が求められる。代数符号帳ゲインは、過去4サブフレームの代数符号帳ゲインの対数エネルギーから予測される予測値 g_c' と γ_c とを用い

て次式により求められる。

$$G_c = g_c' \cdot r_c \quad (21)$$

AMR方式では適応符号帳ゲイン G_a と代数符号帳ゲイン G_c を個別に量子化するので、ゲイン符号変換部85ではAMRの適応符号帳ゲイン量子化器85 b_1 と、代数符号帳ゲイン量子化器85 b_2 とにより個別に量子化を行う。ここで、適応符号帳ゲイン量子化器85 b_1 と、代数符号帳ゲイン量子化器85 b_2 は、AMRで用いられる量子化器と全く同じものである必要はないが、少なくとも、適応符号帳ゲインテーブルと、代数符号帳ゲインテーブルはAMRと同じテーブルを用いるものとする。

【0047】図10は適応符号帳ゲイン量子化器85 b_1 と代数符号帳ゲイン量子化器85 b_2 の構成図である。まず、適応符号帳ゲイン逆量子化値 G_a を適応符号帳ゲイン量子化器85 b_1 に入力してスカラー量子化する。スカラー量子化テーブルSQTaにはAMR方式と同じ16種類(4ビット)の値 $G_a(i)$ ($i=1\sim16$)が記憶されている。二乗誤差演算部ERCaは適応符号帳ゲイン逆量子化値 G_a と各テーブルの値の二乗誤差 $(G_a - G_a(i))^2$ を計算し、インデックス検出部IXDaは i を1~16まで変えたときの誤差が最も小さくなるテーブル値を最適値として求め、そのインデックスをAMR方式における適応符号帳ゲイン符号 $I_GAIN2a(m,0)$ として出力する。次に、雑音符号帳ゲイン逆量子化値 r_c と g_c' から(21)式で求められる G_c を代数符号帳ゲイン量子化器85 b_2 に入力してスカラー量子化する。スカラー量子化テーブルSQTcにはAMR方式と同じ32種類(5ビット)の補正係数の値 $G_c(i)$ ($i=1\sim32$)が記憶されている。二乗誤差演算部ERCcは雑音符号帳ゲイン逆量子化値 G_c と各テーブルの値の二乗誤差 $(G_c - G_c(i))^2$ を計算し、インデックス検出部IXDcは i を1~32まで変えたときの誤差が最も小さくなるテーブル値を最適値として求め、そのインデックスをAMR方式における雑音符号帳ゲイン符号 $I_GAIN2c(m,0)$ として出力する。

【0048】以後、同様の処理を行って、G.729A方式のゲイン符号 $I_GAIN1(n,1)$ からAMR方式の適応符号帳ゲイン符号 $I_GAIN2a(m,1)$ と雑音符号帳ゲイン符号 $I_GAIN2c(m,1)$ を求める。同様に、G.729A方式のゲイン符号 $I_GAIN1(n+1,0)$ からAMR方式の適応符号帳ゲイン $I_GAIN2a(m,2)$ と雑音符号帳ゲイン符号 $I_GAIN2c(m,2)$ を求め、更にG.729A方式のゲイン符号 $I_GAIN1(n+1,1)$ からAMR方式の適応符号帳ゲイン $I_GAIN2a(m,3)$ と雑音符号帳ゲイン符号 $I_GAIN2c(m,3)$ を求める。

(e) 符号送出処理

図3のバッファ部87は、G.729A方式の符号が2フレ

$$LSPc0(i) = 0.75 \cdot old_LSPc(i) + 0.25 \cdot LSPc3(i) \quad (i=1,2,\dots,10) \quad (22)$$

$$LSPc1(i) = 0.50 \cdot old_LSPc(i) + 0.50 \cdot LSPc3(i) \quad (i=1,2,\dots,10) \quad (23)$$

$$LSPc2(i) = 0.25 \cdot old_LSPc(i) + 0.75 \cdot LSPc3(i) \quad (i=1,2,\dots,10) \quad (24)$$

入力音声の性質が有声音のようにあまり急激に変化しない場合にはLSPパラメータの変化も小さい。このため、

ム分(AMR方式の1フレーム分)、処理し終わるまで各変換部82~85より出力する符号を保持し、しかる後、回線データ多重部86に変換された符号を入力する。回線データ多重化部86は、AMR方式の1フレーム分の符号がすべてそろった時点で符号データを多重化して回線データに変換し、出力端子#2より伝送路に送出する。以上説明した通り、第1実施例によればG.729A方式の音声符号を、音声に復号することなしにAMR方式の音声符号に変換することができる。このため、従来のタンデム接続に比べて遅延を少なくでき、しかも、音質の劣化を小さくすることができる。

【0049】(C)第2実施例

図11は本発明の第2実施例の概略説明図である。第2実施例は、第1実施例におけるLSP符号変換部82内のLSP量子化器82bに改良を加えたもので、音声符号変換部の全体構成は第1実施例(図3)と同じである。図11は、G.729A方式の第 n のフレームと第 $(n+1)$ フレームのLSP符号をAMR方式の第 m フレームのLSP符号に変換する場合を示している。図中、 $LSP0(i)$ ($i=1,\dots,10$)はG.729A方式による第 n フレーム/第1サブフレームの10次元LSP逆量子化値、 $LSP1(i)$ ($i=1,\dots,10$)はG.729A方式による第 $(n+1)$ フレーム/第1サブフレームの10次元LSP逆量子化値である。また、 $old_LSP(i)$ ($i=1,\dots,10$)は過去のフレーム(第 $(n-1)$ フレーム目)の10次元LSP逆量子化値である。

【0050】ここで、G.729A方式からAMR方式に音声符号を変換する場合、第1実施例において述べた通り、フレーム長の違いから逆量子化値 $LSP0(i)$ はAMR方式のLSP符号に変換されない。すなわち、G.729A方式で符号化を行う場合には、1フレームにつき1回LSPを量子化するので $LSP0(i)$ 、 $LSP1(i)$ は共に量子化され、復号器側に伝送される。ところが、G.729A方式からAMR方式に音声符号を変換するには、LSPパラメータをAMR方式の復号器の動作に合わせて符号変換する必要がある。このため、G.729A方式の逆量子化値 $LSP1(i)$ はAMR方式の符号に変換されるが逆量子化値 $LSP0(i)$ はAMR方式の符号に変換されない。

【0051】AMR方式では1フレームが4つのサブフレームで構成され、最終サブフレーム(第3サブフレーム)のLSPパラメータのみが量子化されて伝送される。このため、復号器において、第0、第1、第2サブフレームのLSPパラメータ $LSPc0(i)$ 、 $LSPc1(i)$ 、 $LSPc2(i)$ は、前フレームの逆量子化値 $old_LSPc(i)$ と現フレーム/第3サブフレームのLSPパラメータ $LSPc3(i)$ から以下の補間式によって求められる。

第1実施例のように最終サブフレーム(第3サブフレーム)におけるLSP量子化誤差が最小となるようにLSP逆量

子化値を符号に変換し、他の第0～第3サブフレームのLSPパラメータを(22)～(24)式の補間によって求めるようにしても特に問題はない。しかし、無声部や過渡部のように音声の性質が急激に変化する場合には、特にフレーム内で音声の性質が急激に変化する場合には、第1実施例の変換方法では不十分の場合がある。そこで、第2実施例では、最終サブフレームにおけるLSP量子化誤差だけでなく、LSP補間による補間誤差も考慮して符号変換する。

【0052】第1実施例では逆量子化値LSP1(i)をAMR方式のLSP符号に変換する際、該LSP符号より特定されるLSPパラメータLSPc3(i)と逆量子化値LSP1(i)の二乗誤差のみを基準として変換している。これに対して、第2実施例では前記の二乗誤差に加えて、逆量子化値LSP0(i)と(23)式の補間により得られたLSPパラメータLSPc1(i)との間の二乗誤差をも考慮して符号化する。図12は第2実施例のLSP量子化部82bの構成図、図13は第2実施例の変換処理フローであり、10次元の各LSPベクトル(LSPパラメータ)を、低域(1～3次)、中域(4～6次)、高域(7～10次)の3つの小ベクトルに分けて考える。

・低域3次のLSP符号の決定処理

まず、LSP1(i), (i=1, ..., 10)のうち低域小ベクトル(低域の3次)について以下の処理を行う。ただし、ここで用いるLSP符号帳は、低域用符号帳CB1(3次元×512組)、中域用符号帳CB2(3次元×512組)、高域用符号帳CB3(4次元×512組)の3種類である。

【0053】残差ベクトル演算部DBCは低域のLSP逆量子化値LSP1(i) (i=1～3)から予測ベクトルを差引き残差ベクトル $r_1(i)$ (i=1～3)を算出する(ステップ101)。ついで、処理部CPUは $I_1=1$ とし(ステップ102)、低域用符号帳CB1の中から I_1 番目の符号ベクトルCB1(I_1, i) (i=1～3)を取り出し(ステップ103)、該符号ベクトルと残差ベクトル $r_1(i)$ (i=1～3)との間の変換誤差 $E_1(I_1)$ を次式

$$E_1(I_1) = \sum_i \{ r_1(i) - CB1(I_1, i) \}^2 \quad (i=1 \sim 3)$$

を求め、メモリMEMに記憶する(ステップ104)。ついで、処理部CPUは符号ベクトルCB1(I_1, i)を選んだ時のLSP逆量子化値LSPc3(i) (i=1～3)と前回の逆量子化値old_LSPc(i) (i=1～3)とから(23)式によりLSPc1(i) (i=1～3)を補間し(ステップ105)、LSP0(i)とLSPc1(i)との間の変換誤差 $E_2(I_1)$ を次式

$$E_2(I_1) = \sum_i \{ LSP0(i) - LSPc1(i) \}^2 \quad (i=1 \sim 3)$$

により演算し、メモリMEMに記憶する(ステップ106)。

【0054】ついで、処理部CPUは I_1 番目の符号ベクトルを選んだ時の誤差 $E(I_1)$ を次式

$$E(I_1) = E_1(I_1) + E_2(I_1)$$

を演算してメモリに記憶する(ステップ107)。しかる後、誤差 $E(I_1)$ とそれまでの最小誤差 $\min E(I_1)$ とを比較し(ステップ108)、 $E(I_1) < \min E(I_1)$ であれば誤差 $E(I_1)$

を $\min E(I_1)$ に更新する(ステップ109)。更新処理後、処理部は $I_1=512$ になったかチェックし(ステップ110)、 $I_1 < 512$ であれば I_1 を歩進して($I_1+1 \rightarrow I_1$ 、ステップ111)。ステップ103以降の処理を繰り返す。一方、 $I_1=512$ になれば、誤差 $E(I_1)$ が最小となるインデックス I_1 を低域3次のLSP符号として決定する(ステップ112)。

【0055】・中域3次のLPS符号の決定処理

低域3次のLSP符号 I_1 の決定処理が終了すれば、処理部CPUは中域の小ベクトル(3次)について以下の処理を行う。残差ベクトル演算部DBCは中域のLSP逆量子化値LSP1(i) (i=4～6)から予測ベクトルを差引き残差ベクトル $r_2(i)$ (i=4～6)を算出する。ついで、処理部CPUは $I_2=1$ とし、中域用符号帳CB2の中から I_2 番目の符号ベクトルCB2(I_2, i) (i=4～6)を取り出し、該符号ベクトルと残差ベクトル $r_2(i)$ (i=4～6)との間の変換誤差 $E_1(I_2)$ を次式

$$E_1(I_2) = \sum_i \{ r_2(i) - CB2(I_2, i) \}^2 \quad (i=4 \sim 6)$$

を求め、メモリMEMに記憶する。

【0056】ついで、処理部CPUは符号ベクトルCB2(I_2, i)を選んだ時のLSP逆量子化値LSPc3(i) (i=4～6)と前回の逆量子化値old_LSPc(i) (i=4～6)とから(23)式によりLSPc1(i) (i=4～6)を補間し、LSP0(i)とLSPc1(i)との間の変換誤差 $E_2(I_2)$ を次式

$$E_2(I_2) = \sum_i \{ LSP0(i) - LSPc1(i) \}^2 \quad (i=4 \sim 6)$$

により演算し、メモリMEMに記憶する。ついで、処理部CPUは I_2 番目の符号ベクトルを選んだ時の誤差 $E(I_2)$ を次式

$$E(I_2) = E_1(I_2) + E_2(I_2)$$

を演算してメモリに記憶する。しかる後、誤差 $E(I_2)$ とそれまでの最小誤差 $\min E(I_2)$ とを比較し、 $E(I_2) < \min E(I_2)$ であれば誤差 $E(I_2)$ を $\min E(I_2)$ に更新する。更新処理後、処理部は $I_2=512$ になったかチェックし、 $I_2 < 512$ であれば I_2 を歩進して($I_2+1 \rightarrow I_2$)、上記処理を繰り返す。一方、 $I_2=512$ になれば、誤差 $E(I_2)$ が最小となるインデックス I_2 を中域3次のLSP符号として決定する。

【0057】・高域4次のLPS符号の決定処理

中域3次のLSP符号 I_2 の決定処理が終了すれば、処理部CPUは高域の小ベクトル(4次)について以下の処理を行う。残差ベクトル演算部DBCは高域のLSP逆量子化値LSP1(i) (i=7～10)から予測ベクトルを差引き残差ベクトル $r_3(i)$ (i=7～10)を算出する。ついで、処理部CPUは $I_3=1$ とし、高域用符号帳CB3の中から I_3 番目の符号ベクトルCB3(I_3, i) (i=7～10)を取り出し、該符号ベクトルと残差ベクトル $r_3(i)$ (i=7～10)との間の変換誤差 $E_1(I_3)$ を次式

$$E_1(I_3) = \sum_i \{ r_3(i) - CB3(I_3, i) \}^2 \quad (i=7 \sim 10)$$

を求め、メモリMEMに記憶する。ついで、処理部CPUは符号ベクトルCB3(I_3, i)を選んだ時のLSP逆量子化値LSPc3(i) (i=7～10)と前回の逆量子化値old_LSPc(i) (i=7～

10)とから(23)式により $LSPc1(i)$ ($i=7\sim 10$)を補間し、 $LSP0(i)$ と $LSPc1(i)$ との間の変換誤差 $E_2(I_3)$ を次式 $E_2(I_3)=\sum_i \{LSP0(i)-LSPc1(i)\}^2$ ($i=7\sim 10$)により演算し、メモリMEMに記憶する。

【0058】について、処理部CPUは I_3 番目の符号ベクトルを選んだ時の誤差 $E(I_3)$ を次式

$$E(I_3)=E_1(I_3)+E_2(I_3)$$

を演算してメモリに記憶する。しかる後、誤差 $E(I_3)$ とそれまでの最小誤差 $\min E(I_3)$ とを比較し、 $E(I_3)<\min E(I_3)$ であれば誤差 $E(I_3)$ を $\min E(I_3)$ に更新する。更新処理後、処理部は $I_3=512$ になったかチェックし、 $I_3<512$ であれば I_3 を歩進して($I_3+1\rightarrow I_3$)、上記処理を繰り返す。一方、 $I_3=512$ になれば、誤差 $E(I_3)$ が最小となるインデックス I_3 を高域4次のLSP符号として決定する。

【0059】以上、第2実施例では補間部の誤差として $LSPc1(i)$ の変換誤差を考慮したが、同様に $LSPc0(i)$ と $LSPc2(i)$ の変換誤差を考慮してLSP符号を決定することもできる。また、第2実施例では誤差評価基準として E_1 と E_2 の重みが等しいものとして説明したが、 $E=\omega_1 E_1+\omega_2 E_2$ として E_1 と E_2 に別々の重みをかけるようにしてLSP符号を決定することもできる。以上説明した通り、第2実施例によれば、G.729A方式の音声符号を音声に復号することなしにAMR方式の符号に変換することができるため、従来のタンデム接続に比べて遅延を小さくすることができ、しかも、音質劣化も小さくできる。また、 $LSP1(i)$ を再量子化する時の変換誤差だけでなく、LSP補間部による補間誤差も考慮に入れるようにしたから、フレーム内で入力音声の性質が変化するような場合でも変換誤差の少ない良好な音声符号変換を行うことができる。

【0060】(D) 第3実施例

第3実施例は、第2実施例におけるLSP符号変換部82内のLSP量子化器82bに改良を加えたものである。尚、全体の構成図は図3の第1実施例の構成図と同じである。第3実施例では低域小ベクトル、中域小ベクトル、高域小ベクトルごとに予備選択(複数候補の選択)を行い、最後に全帯域での誤差が最小となるLSP符号ベクトルの組み合わせ $\{I_1, I_2, I_3\}$ を決定するところに特徴がある。これは、各帯域で誤差が最小となる符号ベクトルから合成された10次元のLSP合成符号ベクトルが、最適ではない場合が存在するためである。特に、AMR方式やG.729A方式では、10次元のLSPパラメータから変換して得られるLPC係数でLPC合成フィルタを構成しているため、LSPパラメータ領域における変換誤差は再生音声に大きな影響を与える。従って、LSPの小ベクトル単位で誤差が最小になる符号帳探索をするだけでなく、最終的に小ベクトルを結合して得られる10次元のLSPパラメータの誤差(歪)が最小になるようにLSP符号を決定することが望ましい。

【0061】図14、図15は第3実施例のLSP量子化部82bの変換処理フローである。なお、LSP量子化部82bは処理部CPUの処理が異なるだけで図12と同一のブロック構成になっている。LSP逆量子化器82aから出力する10次元の逆量子化値を、低域3次の小ベクトル $LSP1(i)$ ($i=1\sim 3$)、中域3次の小ベクトル $LSP1(i)$ ($i=4\sim 6$)、高域4次の小ベクトル $LSP1(i)$ ($i=7\sim 10$)の3つの小ベクトルに分割する(ステップ201)。ついて、残差ベクトル演算部DBCは低域のLSP逆量子化値 $LSP1(i)$ ($i=1\sim 3$)から予測ベクトルを差引いて残差ベクトル $r_1(i)$ ($i=1\sim 3$)を算出する(ステップ202)。ついて、処理部CPUは $I_1=1$ とし(ステップ203)、低域用符号帳CB1の中から I_1 番目の符号ベクトル $CB1(I_1, i)$ ($i=1\sim 3$)を取り出し(ステップ204)、該符号ベクトルと残差ベクトル $r_1(i)$ ($i=1\sim 3$)との間の変換誤差 $E_1(I_1)$ を次式

$$E_1(I_1)=\sum_i \{r_1(i)-CB1(I_1, i)\}^2 \quad (i=1\sim 3)$$

を求め、メモリMEMに記憶する(ステップ205)。

【0062】について、処理部CPUは符号ベクトル $CB1(I_1, i)$ を選んだ時のLSP逆量子化値 $LSPc3(i)$ ($i=1\sim 3$)と前回の逆量子化値 $old_LSPc(i)$ ($i=1\sim 3$)とから(23)式により $LSPc1(i)$ ($i=1\sim 3$)を補間し(ステップ206)、 $LSP0(i)$ と $LSPc1(i)$ との間の変換誤差 $E_2(I_1)$ を次式

$$E_2(I_1)=\sum_i \{LSP0(i)-LSPc1(i)\}^2 \quad (i=1\sim 3)$$

により演算し、メモリMEMに記憶する(ステップ207)。

ついて、処理部CPUは I_1 番目の符号ベクトルを選んだ時の誤差 $E_L(I_1)$ を次式

$$E_L(I_1)=E_1(I_1)+E_2(I_1)$$

により演算してメモリに記憶する(ステップ208)。しかる後、処理部は $I_1=512$ になったかチェックし(ステップ209)、 $I_1<512$ であれば I_1 を歩進して($I_1+1\rightarrow I_1$ 、ステップ210)。ステップ204以降の処理を繰り返す。一方、 $I_1=512$ になれば、 $E_L(I_1)$ ($I_1=1\sim 512$)のうち小さい方から N_L 個の符号ベクトルの候補を選択し、各候補のインデックスを $PSSEL_{I_1}(j)$ ($j=1, \dots, N_L$)とする(ステップ211)。

【0063】低域3次の小ベクトルについての処理が終了すれば、処理部CPUは中域3次の小ベクトルについて同様の処理を行なう。すなわち、ステップ202～ステップ210と同様の処理により512組の誤差 $E_H(I_2)$ を算出する(ステップ212)。ついて、 $E_H(I_2)$ ($I_2=1\sim 512$)のうち小さい方から N_H 個の符号ベクトルの候補を選択し、各候補のインデックスを $PSSEL_{I_2}(k)$ ($k=1, \dots, N_H$)とする(ステップ213)。中域3次の小ベクトルについての処理が終了すれば、処理部CPUは高域4次の小ベクトルについて同様の処理を行ない、512組の誤差 $E_H(I_3)$ を算出し(ステップ214)、 $E_H(I_3)$ ($I_3=1\sim 512$)のうち小さい方から N_H 個の符号ベクトルの候補を選択し、各候補のインデックスを $PSSEL_{I_3}(m)$ ($m=1, \dots, N_H$)とする(ステップ215)。

【0064】上記の処理により選択された選択候補の中から、以下の処理により全帯域の誤差が最小となる組み

合わせを決定する。すなわち、上記の処理により選ばれた低域 N_L 個、中域 N_M 個、高域 N_H 個のインデックス候補の中から、 $PSEL_{I1}(j)$ 、 $PSEL_{I2}(k)$ 、 $PSEL_{I3}(m)$ を選んだときの合成誤差

$$E(j, k, m) = E_L(PSEL_{I1}(j)) + E_M(PSEL_{I2}(k)) + E_H(PSEL_{I3}(m))$$

を求め(ステップ216)、 j, k, m の全組み合わせの中から合成誤差 $E(j, k, m)$ が最小の組み合わせを決定し、その時のインデックス

$$PSEL_{I1}(j), PSEL_{I2}(k), PSEL_{I3}(m)$$

をAMR方式におけるLSP符号として出力する(ステップ217)。

【0065】尚、第3実施例では補間部の誤差としてLSPC1(i)の変換誤差を考慮したが、LSPC0(i)とLSPC2(i)の変換誤差を考慮してLSP符号を決定することもできる。また、第3実施例では誤差評価基準として E_1 と E_2 の重みが等しいものとして説明したが、 $E = \omega_1 E_1 + \omega_2 E_2$ として E_1 と E_2 に別々の重みをかけるようにしてLSP符号を決定することもできる。

【0066】以上説明した通り、第3実施例によれば、G.729A方式の音声符号を音声に復号することなしにAMR方式の音声符号に変換することができるため、従来のタンドム接続に比べて遅延を小さくすることができ、しかも、音質劣化も小さくできる。また、LSP1(i)を再量子化する時の変換誤差だけでなく、LSP補間部による補間誤差も考慮に入れるようにしたから、フレーム内で入力音声の性質が変化するような場合でも変換誤差の少ない良好な音声符号変換を行うことができる。また、各帯域の複数の符号ベクトルの中から選んだ符号ベクトルの組み合わせのうち、全帯域で合成誤差が最小となる符号ベクトルの組み合わせを求め、該組み合わせに基づいてLSP符号を決定するようにしたから、第2実施例に比べてさらに高い音質の再生音声を提供することができる。

【0067】(E)第4実施例

以上の実施例では符号化方式1としてG.729A符号化方式を用い、符号化方式2としてAMR符号化方式を用いた場合である。第4実施例では、符号化方式1としてAMR方式の7.95kbit/sモードを用い、符号化方式2としてG.729A符号化方式を用いる。図16は第4実施例の音声符号変換部のブロック図であり、図2の原理図と同一部分には同一符号を付している。図2の原理図と異なる点は、バッファ87を設けた点、及び、ゲイン符号変換部85のゲイン逆量子化器を適応符号帳ゲイン逆量子化器85a₁と雑音符号帳ゲイン逆量子化器85a₂で構成している点である。又、図16では、符号化方式1としてAMR方式の7.95kbit/sモードを用い、符号化方式2としてG.729A符号化方式を用いる。

【0068】図16において、AMR方式の符号器(図示せず)から伝送路を介して第mフレーム目の回線データbst1(m)が端子#1に入力する。ここで、AMR方式のピッ

トレートは7.95kbit/s、フレーム長は20msecであるから、回線データbst1(m)は159ビットのビット系列で表される。回線データ分離部81は回線データbst1(m)からLSP符号I_LSP1(m)、ピッチラグ符号I_LAG1(m, j)、代数符号I_CODE1(m, j)、適応符号帳ゲイン符号I_GAIN1a(m, j)、代数符号帳ゲイン符号I_GAIN1c(m, j)を分離して各変換部82~85に入力する。ここで、添字jはAMR方式におけるフレームを構成する4つのサブフレームの番号であり、0, 1, 2, 3のいずれかの値を取る。

【0069】(a) LSP符号変換部

・LSP符号変換処理の概略

図4(b)に示すように、AMR方式のフレーム長は20msecであり、20msecに1回だけ第3サブフレームの入力信号から求めたLSPパラメータを量子化する。これに対し、G.729A方式のフレーム長は10msecであり、10msecに1回だけ第1サブフレームの入力信号から求めたLSPパラメータを量子化する。したがって、AMR方式の1フレーム分のLSP符号からG.729A方式の2フレーム分のLSP符号を作成しなければならない。

【0070】図17は第4実施例におけるLSP符号変換部82の変換処理説明図である。LSP逆量子化器82aは、AMR方式における第mフレーム/第3サブフレームのLSP符号I_LSP1(m)を逆量子化して逆量子化値lspm(i)を発生する。また、この逆量子化値lspm(i)と前フレームである第(m-1)フレーム/第3サブフレームの逆量子化値lspm-1(i)とを用いて補間演算により第mフレーム/第1サブフレームの逆量子化値lspc(i)を予測する。LSP量子化器82bは第mフレーム/第1サブフレームの逆量子化値lspc(i)をG.729A方式に従って量子化して第nフレーム/第1サブフレームのLSP符号I_LSP2(n)を出力する。又、LSP量子化器82bは第mフレーム/第3サブフレームの逆量子化値lspm(i)をG.729A方式に従って量子化してG.729A方式の第(n+1)フレーム/第1サブフレームのLSP符号I_LSP2(n+1)を出力する。

【0071】・LSP逆量子化

図18はLSP逆量子化器82aの構成図である。LSP逆量子化器82aは、AMR方式の10次元のLSPパラメータを1~3次、4~6次、7~10次の3つの小ベクトルに分解した時、各小ベクトルに対して9ビット(512パターン)の符号帳CB1, CB2, CB3を備えている。AMR方式のLSP符号I_LSP1(m)は符号I₁, I₂, I₃に分解されて残差ベクトル算出部DBC入力する。ここで、符号I₁は低域3次の符号帳CB1の要素番号(インデックス)を表し、符号I₂, I₃もそれぞれ中域3次の符号帳CB2, 高域4次の符号帳CB3の要素番号(インデックス)を表す。

【0072】残差ベクトル作成部DBGは、LSP符号I_LSP1(m) = {I₁, I₂, I₃}が与えられると、各符号帳CB1~CB3から符号I₁, I₂, I₃に対応する符号ベクトルを取り出し、以下のように

$$r(i, 1) \sim r(i, 3), r(i, 4) \sim r(i, 6), r(i, 7) \sim r(i,$$

10)

符号帳C B 1～C B 3の順に符号ベクトルを並べて10次元ベクトル $r(i)^{(a)}$ ($i=1, \dots, 10$)を作成する。AMR方式ではLSPパラメータの符号化に際して予測を用いているので $r(i)^{(a)}$ は残差領域のベクトルである。したがって、第 m フレーム目のLSP逆量子化値 $lsp_m(i)$ は、1フレーム前の残差ベクトル $r(i)^{(a-1)}$ に定数 $p(i)$ を乗算して得られるベクトルに現フレームの残差ベクトル $r(i)^{(a)}$ を加えることにより求めることができる。すなわち、逆量子化値算出部R Q Cは次式

$$lsp_a(i) = r(i)^{(a-1)} \cdot p(i) + r(i)^{(a)} \quad (25)$$

$$lsp_c(i) = \frac{lsp_{m-1}(i) + lsp_m(i)}{2}, \quad (i=1, \dots, 10) \quad (26)$$

以上により、LSP逆量子化器8 2 aは第 m フレームにおける第1サブフレーム、第3サブフレームの逆量子化値 $lsp_a(i)$ 、 $lsp_c(i)$ を演算して出力する。

【0073】・LSP量子化

(26)式により補間されたLSPパラメータ $lsp_c(i)$ を以下の方法で量子化することによりG.729A符号化方式の第 n フ

$$\omega(i) = \arccos(lsp_c(i)), \quad (i=1, \dots, 10) \quad (27)$$

により変換する。しかる後、LSF係数 $\omega(i)$ から予測成分(過去4フレームの符号帳出力から得られる予測成分)を減算して得られる残差ベクトルを17ビットで量子化する。

【0074】さて、G.729A符号化方式では3つの符号帳c

$$\tilde{l}_i^{(n-k)} = \begin{cases} cb1(L_1(n-k), i) + cb2(L_2(n-k), i) & (i=1, \dots, 5) \\ cb1(L_1(n-k), i) + cb3(L_3(n-k), i-5) & (i=6, \dots, 10) \end{cases} \quad (28)$$

により求める。ここで、 $L_1(n-k)$ は第 $(n-k)$ フレーム目における符号帳cb1の符号(インデックス)を表し、 $cb1(L_1(n-k))$ は第 $(n-k)$ フレームにおける符号帳cb1のインデックス $L_1(n-k)$ が示す符号ベクトル(出力ベクトル)とする。又、 $L_2(n-k)$ 、 $L_3(n-k)$ についても同様である。ついで、LSF係数 $\omega(i)$ から次式

【数9】

$$l_i = \frac{\omega_i - \sum_{k=1}^4 p(i, k) \tilde{l}_i^{(n-k)}}{1 - \sum_{k=1}^4 p(i, k)} \quad (29)$$

により残差ベクトル l_i ($i=1, \dots, 10$)を求める。ここで、 $p(i, k)$ は予測係数と呼ばれ、あらかじめG.729A符号化方式の規格で定められた定数である。この残差ベクトル l_i がベクトル量子化の対象となる。

【0075】ベクトル量子化は以下のように実行される。まず、符号帳cb1が検索され、二乗平均誤差が最小となる符号ベクトルのインデックス(符号) L_1 を決定する。次に、10次元の残差ベクトル l_i からインデックス L_1 に対応する10次元符号ベクトルを減算し、新たな目標ベクトルを作成する。この新しい目標ベクトルの低次の5次元について符号帳cb2を検索し、二乗平均誤差が最小となる符号ベクトルのインデックス(符号) L_2 を決定す

によりLSP逆量子化値 $lsp_m(i)$ を算出する。尚、 $r(i)^{(a-1)}$ に乘ずる定数 $p(i)$ は、各次数 i 毎にAMR符号化方式の規格により決められているものを使用する。次に、逆量子化値補間部R Q Iは1フレーム前の第 $(m-1)$ フレームで求めてあるLSP逆量子化値 $lsp_{m-1}(i)$ と第 m フレームの $lsp_a(i)$ を用いて第 m フレーム第1サブフレームのLSP逆量子化値 $lsp_c(i)$ を補間演算により求める。補間の方法は任意であるが、例えば次式のような補間方法を用いることができる。

【数7】

フレーム/第1サブフレームに対応するLSP符号 $l_LSP2(n)$ を求めることができる。また、同様の方法で $lsp_a(i)$ を量子化することによりG.729A符号化方式における第 $(n+1)$ フレーム/第1サブフレームに対応するLSP符号 $l_LSP2(n+1)$ を求めることができる。まず、LSP逆量子化値 $lsp_c(i)$ をLSF係数 $\omega(i)$ に次式

$$(27)$$

$b1(10$ 次元、7ビット)、 $cb2(5$ 次元、5ビット)、 $cb3(5$ 次元、5ビット)が設けられている。過去4フレームの各符号帳出力から予測成分 $1^{(n-1)}$ 、 $1^{(n-2)}$ 、 $1^{(n-3)}$ 、 $1^{(n-4)}$ を次式

【数8】

る。同様にして新しい目標ベクトルの高次の5次元について符号帳cb3を検索し、二乗平均誤差が最小となる符号ベクトルのインデックス(符号) L_3 を決定する。以上により求めた符号 L_1 、 L_2 、 L_3 をビット系列として並べてできる17ビットの符号をG.729A符号化方式のLSP符号 $l_LSP2(n)$ として出力する。LSP逆量子化値 $lsp_a(i)$ についても上記と全く同じ方法によりG.729A方式におけるLSP符号 $l_LSP2(n+1)$ を得ることができる。

【0076】図1 9はLSP量子化器8 2 bの構成図であり、残差ベクトル算出部D B Cは(27)～(29)式により残差ベクトルを算出する。第1符号化部C D 1の第1符号帳cb1は10次の符号ベクトルを128組(7ビット)備え、距離演算部D S C 1は残差ベクトル l_i ($i=1 \sim 10$)と符号ベクトル $l(L_1, i)$ ($i=1 \sim 10$)の二乗誤差(ユークリッド距離)を128組演算し、インデックス検出部I X D 1は $L_1=1 \sim 128$ の符号ベクトルのうち、誤差が最小となる符号ベクトルのインデックス L_1 を検出して出力する。減算部S B Cは10次元の残差ベクトル l_i ($i=1 \sim 10$)から第1符号帳cb1のインデックス L_1 に対応する10次元符号ベクトル $l(L_1, i)$ ($i=1 \sim 10$)を減算し、新たな目標ベクトル l_i' ($i=1 \sim 10$)を作成する。この新しい目標ベクトルの低次の5次元ベクトル l_i' ($i=1 \sim 5$)について、第2符号化部C D 2は符号帳cb2

を検索し、二乗平均誤差が最小となる符号ベクトル l'_i (l_2, i) ($i=1\sim5$)のインデックス(符号) l_2 を決定する。同様にして新しい目標ベクトルの高次の5次元ベクトル l'_i ($i=6\sim10$)について、第3符号化部 CD 3は符号帳cb3を検索し、二乗平均誤差が最小となる符号ベクトル l'_i (l_3, i) ($i=6\sim10$)のインデックス(符号) l_3 を決定する。

【0077】(b) ビッチラグ符号変換部

次に、ビッチラグ符号の変換について説明する。第1実施例で説明した通り、G.729A符号化方式及びAMR符号化方式では、サンプル補間フィルタを使用して1/3サンプル精度でビッチラグを決定している。このため、ビッチラグは整数ラグと非整数ラグの2種類が存在する。G.729A方式におけるビッチラグとインデックスの関係は図7に示す通りであり、第1実施例と同じであるので詳しい説明を省略する。また、AMR方式におけるビッチラグとインデックスの関係は図8の通りであり、これも第1実施例と同じであるので詳しい説明を省略する。

【0078】したがって、偶数サブフレームについてはAMR方式とG.729A方式とではビッチラグの量子化方法と量子化ビット数が全く同じである。このため、AMR方式の偶数サブフレームのビッチラグインデックスを次式

$$I_LAG2(n,0)=I_LAG1(m,0) \quad (30)$$

$$I_LAG2(n+1,0)=I_LAG1(m,2) \quad (31)$$

によりG.729A方式の連続する2つのフレームの第0サブフレームのビッチラグインデックスに変換することができる。一方、奇数サブフレームについては、前サブフレームの整数ラグToldと現サブフレームのビッチラグの差分を量子化する点で共通であるが、AMR方式の量子化ビット数(6bit)に対してG.729A方式の量子化ビット数(5bit)が少ない。このため、以下のような工夫が必要である。

【0079】まず、AMR方式の第mフレーム／第1サブフレームのラグ符号 $I_LAG1(m,1)$ から整数ラグ $Int(m,1)$ と非整数ラグ $Frac(m,1)$ を求め、ビッチラグを次式 $P=Int(m,1)+Frac(m,1)$ により求める。インデックス(ラグ符号)に対応する整数ラグと非整数ラグは一対一に対応しているので、図8(b)に示すように例えばラグ符号が28であれば整数ラグは-1、非整数ラグは-1/3となり、ビッチラグPは $-(1+1/3)$ となる。次に、求めたビッチラグPが図7(b)に示すG.729A方式の奇数サブフレームにおける5ビットのビッチラグ範囲Told- $(5+2/3)$ ～Told+ $(4+2/3)$ に入るかどうかを調べる。以下では、このビッチラグ範囲を{Told- $(5+2/3)$, Told+ $(4+2/3)$ }と表すことにする。第1実施例で説明したように、奇数サブフレームでは、AMR方式のビッチラグとインデックスの対応関係とG.729A方式のビッチラグとインデックスの対応関係を比較すると、インデックスが15だけずれている。したがって、ビッチラグPが前記ビッチラグ範囲に入っている場合は、次式

$$I_LAG2(n,1)=I_LAG1(m,1)-15 \quad (32)$$

$$I_LAG2(n+1,1)=I_LAG1(m,3)-15 \quad (33)$$

により補正する。これにより、AMR方式のビッチラグ符号 $I_LAG1(m,1)$ をG.729A方式のビッチラグ符号 $I_LAG2(n,1)$ に変換することができる。また、同様にしてAMR方式のビッチラグ符号 $I_LAG1(m,3)$ をG.729A方式のビッチラグ符号 $I_LAG2(n+1,1)$ に変換できる。

【0080】また、ビッチラグPが前記ビッチラグ範囲に入っていない場合にはビッチラグをクリップ処理する。つまり、ビッチラグPがTold- $(5+2/3)$ よりも小さい場合には、例えばTold-7の場合には、ビッチラグPをTold- $(5+2/3)$ にクリップする。また、ビッチラグPがTold+ $(4+2/3)$ よりも大きい場合には、例えばTold+7の場合にはビッチラグPをTold+ $(4+2/3)$ にクリップする。一見すると、このようなビッチラグのクリップは音声品質の劣化を招くように見えるが、発明者等による予備実験によると前記のクリップ処理を行っても音質はほとんど劣化しないことが確認された。一般に、「ア」や「イ」などの有声部ではビッチラグは滑らかに変化することが知られており、有音部の奇数サブフレームにおけるビッチラグPの変動は小さく、ほとんどの場合、{Told- $(5+2/3)$, Told+ $(4+2/3)$ }に入っている。一方、立ち上りや立ち下がりなどの変動部では、ビッチラグPの値が上記の範囲を超えることがあるが、音声の性質が変化する部分では適応符号帳による周期性音源の再生音声に対する影響が小さくなるため、前述したクリップ処理を行っても音質にはほとんど影響がない。以上説明した方法によれば、AMR方式のビッチラグ符号をG.729A方式のビッチラグ符号に変換することができる。

【0081】(c) 代数符号変換

次に、代数符号の変換について説明する。AMR方式とG.729A方式とではフレーム長が異なるが、サブフレーム長は5msec(40サンプル)で共通であり、代数符号の構造は両方式で全く同じ構造となっている。したがって、AMR方式の代数符号帳探索の出力結果である4本のパルス位置とパルスの極性情報は、そのままG.729A方式の代数符号帳出力結果と一対一で置き換えることが可能である。したがって、代数符号の変換式は次式となる。

$$I_CODE2(n,0)=I_CODE1(m,0) \quad (34)$$

$$I_CODE2(n,1)=I_CODE1(m,1) \quad (35)$$

$$I_CODE2(n+1,0)=I_CODE1(m,2) \quad (36)$$

$$I_CODE2(n+1,1)=I_CODE1(m,3) \quad (37)$$

【0082】(d) ゲイン符号変換

次にゲイン符号の変換について説明する。まず、AMR方式の第mフレーム／第0サブフレームの適応符号帳ゲイン符号 $I_GAIN1a(m,0)$ を適応符号帳ゲイン逆量子化器85a₁に入力して適応符号帳ゲインの逆量子化値Gaを求める。適応符号帳ゲイン逆量子化器85a₁はAMR方式のそれと同じ4ビット(16パターン)の適応符号帳ゲインテーブルを備えており、該テーブルを参照して符号 I_GAIN

1a(m,0)に対応する適応符号帳ゲインの逆量子化値Gaを出力する。

【0083】次に、AMR方式の第mフレーム/第0サブフレームの代数符号帳ゲイン符号I_GAIN1c(m,0)を代数符号帳ゲイン逆量子化器85a₂に入力して代数符号帳ゲインの逆量子化値G_cを求める。AMR方式では代数符号帳ゲインの量子化にフレーム間予測が用いられており、過去の4サブフレームの代数符号帳ゲインの対数エネルギーからゲインを予測し、その補正係数を量子化する。このため、代数符号帳ゲイン逆量子化器85a₂はAMR方式のそれと同じ5ビット(32パターン)の補正係数テーブルを備え、符号I_GAIN1c(m,0)に対応する補正係数のテーブル値r_cを求め、代数符号帳ゲインの逆量子化値G_c=(g_c'×r_c)を出力する。尚、ゲイン予測方法はAMR方式の復号器で行う予測方法と全く同じである。

【0084】次に、ゲインGa、G_cをゲイン量子化器85bに入力してG.729A方式のゲイン符号に変換する。ゲイン量子化器85bではG.729A方式のそれと同じ7ビットのゲイン量子化テーブルを使用する。この量子化テーブルは2次元であり、1番目の要素は適応符号帳ゲインGaであり、2番目の要素は代数符号帳ゲインに対応する補正係数r_cである。ここで、G.729A方式では代数符号帳ゲインの量子化にフレーム間予測を用いており、その予測方法はAMR方式と同じである。第4実施例では、AMR方式の各符号I_LAG1(m,0)、I_CODE1(m,0)、I_GAIN1a(m,0)、I_GAIN1c(m,0)から各逆量子化器82a~85a₂によって得られた逆量子化値を用いてAMR側の音源信号を求め、これを参照用音源信号とする。

【0085】次に、すでにG.729A方式に変換されたピッチラグ符号I_LAG2(n,0)からピッチラグを求め、このピッチラグに対応する適応符号帳出力を求める。また、変換された代数符号I_CODE2(n,0)から代数符号帳出力を作成する。しかる後、G.729A用のゲイン量子化テーブルからインデックス順に1組ずつテーブル値を取り出して適応符号帳ゲインGaと代数符号帳ゲインG_cを求める。ついで、これら適応符号帳出力、代数符号帳出力、適応符号帳ゲイン、代数符号帳ゲインとからG.729A方式に変換した時の音源信号(試験用音源信号)を作成し、前記参照用音源信号と試験用音源信号の誤差電力を計算する。全てのインデックスが示すゲイン量子化テーブル値について同様の処理を行い、最も誤差電力が小さくなるインデックスを最適なゲイン量子化符号とする。

【0086】具体的な処理手順を以下に示す。

(1) まず、AMR方式のピッチラグ符号I_LAG1に対応する適応符号帳出力pitch₁(i)(i=0,1,...,39)を求める。

(2) 参照用音源信号を次式

$$ex_1(i) = G_a \cdot pitch_1(i) + G_c \cdot code(i) \quad (i=0,1,\dots,39)$$

を求める。

(3) G.729A方式のピッチラグI_LAG2(n,k)に対応する適

応符号帳出力pitch₂(i)(i=0,1,...,39)を求める。

(4) ゲイン量子化テーブルからL番目のゲイン符号に対応するテーブル値Ga₂(L)、r_c(L)を取り出す。

(5) 過去のサブフレームの代数符号帳ゲインから予測されるエネルギー成分g_c'を計算し、G_{c2}(L)=g_c'・r_c(L)を求める。

【0087】(6) 試験用音源信号を次式

$$ex_2(i,L) = Ga_2(L) \cdot pitch_2(i) + G_{c2}(L) \cdot code(i) \quad (i=0,1,\dots,39)$$

を求める。尚、代数符号帳出力code(i)はAMR方式とG.729A方式とで等しい。

(7) 二乗誤差を次式

$$E(L) = (ex_1(i) - ex_2(i,L))^2 \quad (i=0,1,\dots,39)$$

を求める。

(8) E(L)をゲイン量子化テーブルの全インデックスのパターン(L=0~127)について計算し、E(L)が最も小さくなるLを最適なゲイン符号I_GAIN2(n,0)として出力する。

以上では、最適なゲイン符号を探索する際に、音源信号の二乗誤差を基準として用いたが、音源信号から再生音声を求め、再生音声領域でゲイン符号を探索する構成としてもよい。

【0088】(e) 符号送出処理

AMR方式とG.729A方式とではフレーム長が異なるため、AMR方式の1フレーム分の回線データからG.729A方式の2フレーム分の回線データが得られる。このため、バッファ部87(図16)は、まず、符号I_LSP2(n), I_LAG2(n,0), I_LAG2(n,1), I_CODE2(n,0), I_CODE2(n,1), I_GAIN2(n,0), I_GAIN2(n,1)を回線データ多重化部86へ入力する。回線データ多重化部86は入力符号を多重してG.729Aの第nフレームの音声符号を作成し、回線データとして伝送路に送出する。ついで、バッファ部87は符号I_LSP2(n+1), I_LAG2(n+1,0), I_LAG2(n+1,1), I_CODE2(n+1,0), I_CODE2(n+1,1), I_GAIN2(n+1,0), I_GAIN2(n+1,1)を回線データ多重化部86に入力する。回線データ多重化部86は入力符号を多重してG.729Aの第(n+1)フレームの音声符号を作成し、回線データとして伝送路に送出する。

【0089】(F) 第5実施例

以上の実施例では、伝送路誤りがない場合である。実際には、携帯電話のように無線通信を用いる場合、フェージング等の影響によりビット誤りやバースト誤りが発生し、音声符号が本来と異なるものに変化したり、1フレーム全部の音声符号が欠落してしまう場合がある。また、インターネットでは網が混雑していると伝送遅延が大きくなり、1フレーム全部の音声符号が欠落したり、フレームの順番が入れ替わってしまう場合がある。

【0090】(a) 伝送路誤りの影響

図20は伝送路誤りの影響を説明する説明図であり、図1、図2と同一部分には同一符号を付している。異なる

点は、伝送路誤り(回線誤り)を伝送信号にモデル的に加える合成部95を有している点である。入力音声は符号化方式1の符号器61aに入力され符号化方式1の音声符号V1が出力される。音声符号V1は無線回線または有線回線(インターネット等)の伝送路71を通過して音声符号変換部80に入力される。ただし、音声符号変換部80に入力される前に回線誤りが混入すると音声符号V1は回線誤りの影響により音声符号V1と異なった音声符号V1'に変形される。音声符号V1'は符号分離部81に入力され、LSP符号、ピッチラフ符号、代数符号、ゲイン符号の各パラメータ符号に分離され、各符号変換部82~85で符号化方式2に適した符号に変換される。変換された各符号は符号多重部86で多重され、最終的に符号化方式2の音声符号V2が出力される。

【0091】以上のように、音声符号変換部80に入力する前に回線誤りが混入すると、誤った音声符号V1'を基に変換が行われるため、変換された音声符号V2は必ずしも最適な符号とはならない。また、CELPでは音声合成フィルタとしてIIRフィルタを用いているため、回線誤りの影響でLSP符号やゲイン符号等が最適でない場合には、しばしばフィルタが発振して大きな異音が発生する可能性がある。また、IIRフィルタの性質上、一度フィルタが発振すると後続フレームへも影響を及ぼすと言う問題がある。このため、回線誤りによる音声符号変換部経の影響を小さくする必要がある。

【0092】(b) 第5実施例の原理

図21は第5実施例の原理図であり、符号化方式1と符号化方式2としてAMRやG.729AなどのCELPをベースとした符号化方式が用いられる。図21において、入力音声xinが符号化方式1の符号器61aへ入力され符号化方式1の音声符号sp1を発生する。音声符号sp1は、無線回線又は有線回線(インターネット等)の伝送路71を通過して音声符号変換部80へ入力する。ここで、音声符号変換部80に入力される前に回線誤りERRが混入すると、音声符号sp1は回線誤りの入った音声符号sp'に変形される。回線誤りERRのパターンはシステムに依存し、ランダムビット誤り、バースト性誤りなどの様々なパターンを取り得る。尚、誤りが混入しない場合にはsp1'とsp1は全く同じ符号となる。

【0093】音声符号sp1'は符号分離部81へ入力され、LSP符号LSP1、ピッチラフ符号Lag1、代数符号PCB1、ゲイン符号Gain1に分離される。又、音声符号sp1'は回線誤り検出部96に入力し、周知の方法で回線誤りの有無が検出される。たとえば音声符号sp1にCRC符号を付加したり、フレーム順序を示すデータを付加しておくことにより回線誤りを検出することができる。LSP符号LSP1はLSP修正部82cに入力され、回線誤りの影響が軽減されたLSP符号LSP1'に変換される。ピッチラフ符号Lag1はピッチラフ修正部83cに入力され、回線誤りの影

響が軽減されたピッチラフ符号Lag1'に変換される。代数符号PCB1は代数符号修正部84cに入力され、回線誤りの影響が軽減された代数符号PCB1'に変換される。ゲイン符号Gain1はゲイン符号修正部85cに入力され、回線誤りの影響が軽減されたゲイン符号Gain1'に変換される。

【0094】次に、LSP符号LSP1'はLSP符号変換部82に入力されて符号化方式2のLSP符号LSP2に変換される。ピッチラフ符号Lag1'はピッチラフ符号変換部83に入力されて符号化方式2のピッチラフ符号Lag2に変換される。代数符号PCB1'は代数符号変換部84に入力されて符号化方式2の代数符号PCB2に変換される。ゲイン符号Gain1'はゲイン符号変換部85に入力されて符号化方式2のゲイン符号Gain2に変換される。各符号LSP2、Lag2、PCB2、Gain2は符号多重化部86で多重化され、符号化方式2の音声符号sp2が出力される。以上のような構成にすることにより、従来の音声符号変換部で問題となっていた回線誤りの影響による変換後の音声品質の劣化を小さくすることができる。

【0095】(c) 第5実施例の音声符号変換部

図22は第5実施例の音声符号変換部の構成図であり、符号化方式1としてG.729A、符号化方式2としてAMRを用いた場合を示している。尚、AMRには8つの符号化モードが存在するが、ここでは7.95kbit/sを用いた場合を示している。図22において、第nフレーム目のG.729Aの符号器出力である音声符号sp1(n)が音声符号変換部80に入力される。G.729Aのビットレートは8kbit/sであるからsp1(n)は80ビットのビット系列で表される。符号分離部81は、音声符号sp1(n)をLSP符号LSP1(n)、ピッチラフ符号Lag1(n,j)、代数符号PCB1(n,j)、ゲイン符号Gain1(n,j)に分離する。括弧内の添字jはサブフレーム番号を表し0,1の値をとる。

【0096】ここで、音声符号sp1(n)が音声符号変換部80に入力される前に回線誤りERRが混入すると、音声符号sp1(n)は回線誤りの入った音声符号sp1'(n)に変形される。回線誤りERRのパターンはシステムに依存し、ランダムビット誤りやバースト性誤りなどの様々なパターンを取り得る。バースト性の誤りがある場合には、フレーム全体の情報が失われてしまうので音声を正しく再生することができない。また、ネットワークの混雑などによりあるフレームの音声符号が所定の時間内に届かない場合は、そのフレームはなかったものとして扱われるため、フレーム全体の情報が失われてしまい音声の正しく再生することができない。これをフレーム消失と呼び、回線誤りと同様に何らかの対策が必要である。尚、誤りが混入しない場合にはsp1'(n)とsp1(n)は全く同じ符号となる。

【0097】回線誤りの有無、フレーム消失の有無の判定方法はシステムによって異なるが、例えば、携帯電話システムの場合には、音声符号に誤り検出符号と誤り訂

正符号を付加することが一般的であるので、回線誤り検出部96はこの誤り検出符号により現フレームの音声符号に誤りがあるかないかを検出することができる。また、所定の時間内に1フレーム分の音声符号が全て受信できなかった場合には、そのフレームはフレーム消失として扱うことができる。LSP符号LSP1(n)はLSP符号修正部82cに入力され、回線誤りの影響が軽減されたLSPパラメータ $l_{sp}(i)$ に変換される。ピッチラグ符号 $lag1(n, j)$ はピッチラグ修正部83cに入力され、回線誤りの影響が軽減されたピッチラグ符号 $lag1'(n, j)$ に変換される。代数符号PCB1(n, j)は代数符号修正部84cに入力され、回線誤りの影響が軽減された代数符号 $PCB1'(n, j)$ に変換される。ゲイン符号 $Gain1(n, j)$ はゲイン符号修正部85cに入力され、回線誤りの影響が軽減されたピッチゲイン $Ga(n, j)$ と代数符号帳ゲイン $Gc(n, j)$ に変換される。

【0098】尚、回線誤り又はフレーム消失が発生しなければ、LSP符号修正部82cは第1実施例と同一のLSPパラメータ $l_{sp}(i)$ が出力され、ピッチラグ修正部83c

から $lag1(n, j)$ と全く同じ符号が $lag1'(n, j)$ として出力され、代数符号修正部84cから $PCB1(n, j)$ と全く同じ符号が $PCB1'(n, j)$ として出力され、ゲイン修正部85cから第1実施例と同一のピッチゲイン $Ga(n, j)$ と代数符号帳ゲイン $Gc(n, j)$ が出力される。

【0099】(d) LSP符号修正及びLSP符号変換
LSP修正部82cについて説明する。誤りのないLSP符号LSP1(n)がLSP修正部82cに入力されると、LSP修正部82cは第1実施例のLSP逆量子化器82aと同一の処理を行う。すなわち、LSP修正部82cではLSP1(n)を L_0, L_1, L_2, L_3 の4つの小符号に分割する。ここで、 L_1 はLSP符号帳CB1の要素番号を表し、 L_2, L_3 はそれぞれLSP符号帳CB2、CB3の要素番号を表す。CB1は10次元のベクトルを128組持ち、CB2とCB3は共に5次元ベクトルを32組持つ符号帳である。尚、 L_0 は後述する2種類のMA予測係数のうちどちらを使うかを表す。第nフレーム目の残差ベクトル $l_i^{(n)}$ を次式

【数10】

$$l_i^{(n)} = \begin{cases} CB1(L_1, i) + CB2(L_2, i) & (i = 1, \dots, 5) \\ CB1(L_1, i) + CB3(L_3, i - 5) & (i = 6, \dots, 10) \end{cases} \quad (38)$$

により求める。

【0100】ついで、残差ベクトル $l_i^{(n)}$ と最新の過去4フレームで求めた残差ベクトル $l_i^{(n-k)}$ を用いてLSF係

数 $\omega(i)$ を次式

【数11】

$$\omega(i) = (1 - \sum_{k=1}^4 p(i, k)) l_i^{(n)} + \sum_{k=1}^4 p(i, k) l_i^{(n-k)}, \quad (i = 1, \dots, 10) \quad (39)$$

より求める。ここで、 $p(i, k)$ は2種類のMA予測係数のうち L_0 により指定された方の係数を表す。 $l_i^{(n)}$ は次フレーム以降のためにバッファ82dに保持される。しかる後、次式によりLSF係数 $\omega(i)$ からLSPパラメータ $l_{sp}(i)$ を求める。この $l_{sp}(i)$ がLSP修正部82cの出力となる。

$$l_{sp}(i) = \cos(\omega(i)), \quad (i = 1, \dots, 10) \quad (40)$$

【0101】以上のように、回線誤りやフレーム消失がない場合には、前記の方法により現フレームで受信した

LSP符号と、過去4フレームで受信したLSP符号とからLSPパラメータを算出することにより、LSP符号変換部82への入力を作成することができる。一方、回線誤りやフレーム消失により現フレームの正しいLSP符号を受信できない場合に上記の手順を用いることができない。そこで第5実施例においてLSP修正部82cは回線誤りやフレーム消失がある場合、最後に受信した良好な過去4フレームのLSP符号から残差ベクトル $l_i^{(n)}$ を(41)式

【数12】

$$l_i^{(n)} = \left[\hat{\omega}_i^{(m)} - \sum_{k=1}^4 p(i, k) l_i^{(n-k)} \right] / \left[1 - \sum_{k=1}^4 p(i, k) \right], \quad (i = 1, \dots, 10) \quad (41)$$

により作成する。ここで、 $p(i, k)$ は最後に受信した良好なフレームのMA予測係数である。

【0102】以上の通り、回線誤りやフレーム消失により現フレームの音声符号を受信できない場合でも第5実施例では(41)式により現フレームの残差ベクトル $l_i^{(n)}$ を求めることができる。LSP符号変換部82は第1実施例のLSP量子化器82bと同一の処理を行う。すなわち、LSP修正部82cからのLSPパラメータ $l_{sp}(i)$ を入力として第1実施例と同一の逆量子化処理を行って、AMR用

のLSP符号を求める。

【0103】(e) ピッチラグ修正及びピッチラグ符号変換

次にピッチラグ修正部83cについて説明する。ピッチラグ修正部83cでは、回線誤りやフレーム消失しなければ、受信した現フレームのラグ符号を $lag1'(n, j)$ として出力する。また、回線誤りやフレーム消失があれば、バッファ83dに記憶してある最後に受信した良好なフレームのピッチラグ符号を $lag1'(n, j)$ として出力

するように動作する。一般的に、有声部ではピッチラグが滑らかに変化することが知られている。したがって、有声部では上記のように前フレームのピッチラグで代用させても音質上の劣化はほとんどない。また、無声部ではピッチラグは大きく変換することが知られているが、無声部における適応符号帳の寄与率は小さい(ピッチゲインが小さい)ため、前述の方法による音質劣化はほとんどない。

【0104】ピッチラグ変換部83では、第1実施例と同一のピッチラグ符号変換を行う。すなわち、G.729Aはフレーム長が10ミリ秒であるのに対して、AMRのフレーム長は20ミリ秒であることから、ピッチラグ符号を変換するにはG.729Aの2フレーム分のピッチラグ符号をAMRの1フレーム分のピッチラグ符号として変換する必要がある。今、G.729Aの第nフレーム目と第n+1フレーム目のピッチラグ符号をAMRの第mフレーム目のピッチラグ符号に変換する場合を考える。ピッチラグ符号は、整数ラグと非整数ラグとを1語に合成したものである。図6の偶数サブフレームでは、G.729AとAMRにおけるピッチラグ符号の合成方法は全く同じであり、その量子化ビット数も同じ8ビットであることから次式

$$\text{LAG2}(m,0)=\text{LAG1}'(n,0) \quad (42)$$

$$\text{LAG2}(m,2)=\text{LAG1}'(n+1,0) \quad (43)$$

のようにピッチラグ符号を変換することができる。

【0105】また、奇数サブフレームについては、前サブフレームの整数ラグとの差分量子化を行う点では共通であるが、量子化ビット数がAMRの方が1ビット多いことから、次式

$$\text{LAG2}(m,1)=\text{LAG1}'(n,1)+15 \quad (44)$$

$$\text{LAG2}(m,3)=\text{LAG1}'(n+1,1)+15 \quad (45)$$

のように変換することができる。

【0106】(f) 代数符号修正及び代数符号変換
代数符号修正部84cでは、回線誤りやフレーム消失がない場合、受信した現フレームの代数符号をPCB1'(n,j)として出力する。また、回線誤りやフレーム消失があった場合には、バッファ84dに記憶してある最後に受信した良好なフレームの代数符号をPCB1'(n,j)として出力するように動作する。代数符号変換部84では、第1実施例と同様の代数符号変換を行う。すなわち、G.729AとAMRとではフレーム長が異なるが、サブフレーム長は5ミリ秒(40サンプル)で共通である。また、代数符号の構造についても両方式では全く同じ構造となっている。したがって、G.729Aの代数符号帳探索の出力結果であるパルス位置とパルスの極性情報は、そのままAMRの代数符号帳出力結果と一対一で置き換えることが可能である。したがって、代数符号の変換式は次式

$$\text{PCB2}(m,0)=\text{PCB1}'(n,0) \quad (46)$$

$$\text{PCB2}(m,1)=\text{PCB1}'(n,1) \quad (47)$$

$$\text{PCB2}(m,2)=\text{PCB1}'(n+1,0) \quad (48)$$

$$\text{PCB2}(m,3)=\text{PCB1}'(n+1,1) \quad (49)$$

となる。

【0107】(g) ゲイン符号修正及びゲイン符号変換
ゲイン符号修正部85cでは、回線誤りやフレーム消失がない場合、第1実施例と同様に、受信した現フレームのゲイン符号Gain1(n,j)からピッチゲインGa(n,j)と代数符号帳ゲインGc(n,j)を求める。ただし、G.729Aでは代数符号帳ゲインをそのまま量子化するのではなく、ピッチゲインGa(n,j)と代数符号帳ゲインに対する補正係数 γ_c を組み合わせてベクトル量子化する。したがって、ゲイン修正部85cはゲインGain1(n,j)が入力すると、G.729Aのゲイン量子化テーブルからゲインGain1(n,j)に対応するピッチゲインGa(n,j)と補正係数 $\gamma_c(n,j)$ を求める。次に、補正係数 $\gamma_c(n,j)$ と過去4サブフレームの代数符号帳ゲインの対数エネルギーから予測される予測値 g_c' を用いて(21)式により代数符号長ゲインGc(n,j)を求める。

【0108】回線誤りやフレーム消失がある場合には、現フレームのゲイン符号を用いることができないので、(50)~(53)式のようにバッファ85d₁, 85d₂に記憶してある1サブフレーム前のゲインを減衰してピッチゲインGa(n,j)と代数符号帳ゲインGc(n,j)を求める。ここで α 、 β は1以下の定数である。このピッチゲインGa(n,j)と代数符号帳ゲインGc(n,j)がゲイン修正部85cの出力となる。

$$\text{Ga}(n,0)=\alpha \cdot \text{Ga}(n-1,1) \quad (50)$$

$$\text{Ga}(n,1)=\alpha \cdot \text{Ga}(n,0) \quad (51)$$

$$\text{Gc}(n,0)=\beta \cdot \text{Gc}(n-1,1) \quad (52)$$

$$\text{Gc}(n,1)=\beta \cdot \text{Gc}(n,0) \quad (53)$$

次にゲイン変換部85b₁', 85b₂'について説明する。AMRではピッチゲインと代数符号帳ゲインを個別に量子化する。ただし、代数符号帳ゲインは直接には量子化されず、代数符号帳ゲインに対する補正係数が量子化される。まず、ピッチゲインGa(n,0)をピッチゲイン変換部85b₁'に入力しスカラー量子化する。このスカラー量子化テーブルにはAMRと同じ16種類(4ビット)の値が記憶されている。量子化の方法は、ピッチゲインGa(n,0)と各テーブル値の自乗誤差を計算し、その誤差が最も小さくなるテーブル値を最適値とし、そのインデックスをGain2a(m,0)とする。

【0109】代数符号帳ゲイン変換部85b₂'では、 $\gamma_c(n,0)$ をスカラー量子化する。このスカラー量子化テーブルにはAMRと同じ32種類(5ビット)の値が記憶されている。量子化の方法は、 $\gamma_c(n,0)$ と各テーブル値の自乗誤差を計算し、その誤差も最も小さくなるテーブル値を最適値とし、そのインデックスをGain2c(m,0)とする。同様の処理を行って、Gain1(n,1)からGain2a(m,1)とGain2c(m,1)を求める。また、Gain1(n+1,0)からGain2a(m,2)とGain2c(m,2)を求め、Gain1(n+1,1)からGain2a(m,3)とGain2c(m,3)を求める。

【0110】(h) 符号多重

符号多重化部86では、G.729Aの符号を2フレーム分(AMRの1フレーム分)を処理し終わるまで変換された符号を保持し、G.729Aの符号を2フレーム処理してAMRの1フレーム分の符号が全てそろった段階で音声符号s_{p2(m)}を出力するように動作する。以上説明した通り、本実施例によれば回線誤りやフレーム消失がある場合に、G.729Aの音声符号をAMRの符号に変換する際の誤り影響を小さくすることができるため、従来の音声符号変換部に比べて音質の劣化も少ない良好な音声品質を実現することができる。

【0111】・付記

(付記1) 第1の音声符号化方式により符号化して得られる音声符号を入力され、該音声符号を第2の音声符号化方式の音声符号に変換して出力する音声符号変換装置において、第1の音声符号化方式による音声符号より、音声信号を再現するために必要な複数の成分の符号を分離する符号分離手段、各成分の符号をそれぞれ逆量子化して逆量子化値を出力する逆量子化部、前記各逆量子化部から出力する逆量子化値を第2の音声符号化方式により量子化して符号を発生する量子化部、各量子化部から出力する符号を多重して第2の音声符号化方式による音声符号を出力する手段、を備えたことを特徴とする音声符号変換装置。

【0112】(付記2) 音声信号の一定サンプル数をフレームとし、フレーム毎の線形予測分析により得られる線形予測係数(LPC係数)又は該LPC係数から求めるLSPパラメータを量子化することにより得られる第1のLPC符号と、周期性音源信号を出力するための適応符号帳の出力信号を特定する第1のピッチラグ符号と、雑音性音源信号を出力するための雑音符号帳の出力信号を特定する第1の雑音符号と、前記適応符号帳の出力信号の振幅を表す適応符号帳ゲインと前記雑音符号帳の出力信号の振幅を表す雑音符号帳ゲインとを量子化して得られる第1のゲイン符号とを求め、これら符号で音声信号を符号化する方式を第1の音声符号化方式とし、第1の音声符号化方式と異なる量子化方法により量子化して得られる第2のLPC符号、第2のピッチラグ符号、第2の雑音符号、第2のゲイン符号とで音声信号を符号化する方式を第2の音声符号化方式とすると、第1の音声符号化方式により符号化した音声符号を入力され、該音声符号を第2の音声符号化方式の音声符号に変換する音声符号変換装置において、前記第1のLPC符号を第1の音声符号化方式のLPC逆量子化方法により逆量子化し、得られるLPC係数の逆量子化値を第2の音声符号化方式のLPC量子化テーブルを用いて量子化して第2のLPC符号を求めるLPC符号変換手段、第1の音声符号化方式におけるピッチラグ符号と第2の音声符号化方式におけるピッチラグ符号との相違を考慮した変換処理により、前記第1のピッチラグ符号を第2のピッチラグ符号に変換するピッチラグ変換手段、第1の音声符号化方式における雑音符号と

第2の音声符号化方式における雑音符号との相違を考慮した変換処理により、前記第1の雑音符号を第2の雑音符号に変換する雑音符号変換手段、前記第1のゲイン符号を第1の音声符号化方式のゲイン逆量子化方法により逆量子化してゲイン逆量子化値を求めるゲイン逆量子化手段、前記ゲイン逆量子化値を第2の音声符号化方式のゲイン量子化テーブルを用いて量子化して第2のゲイン符号に変換するゲイン符号変換手段、とを有することを特徴とする音声符号変換装置。

【0113】(付記3) 前記ゲイン逆量子化手段は、前記第1のゲイン符号を第1の音声符号化方式のゲイン逆量子化方法により逆量子化して適応符号帳ゲインの逆量子化値と雑音符号帳ゲインの逆量子化値を求め前記ゲイン符号変換手段は、前記適応符号帳ゲイン及び雑音符号帳ゲインの逆量子化値をそれぞれ個別に第2の音声符号化方式のゲイン量子化テーブルを用いて量子化して適応符号帳ゲイン符号及び雑音符号帳ゲイン符号を発生し、これら2つのゲイン符号で前記第2のゲイン符号を構成する、ことを特徴とする付記2記載の音声符号変換装置。

【0114】(付記4) 前記ゲイン符号変換手段は、前記適応符号帳ゲインの逆量子化値を第2の音声符号化方式のゲイン量子化テーブルを用いて量子化して適応符号帳ゲイン符号を発生する第1ゲイン符号変換手段、前記雑音符号帳ゲインの逆量子化値を第2の音声符号化方式のゲイン量子化テーブルを用いて量子化して雑音符号帳ゲイン符号を発生する第2ゲイン符号変換手段、を有することを特徴とする付記3記載の音声符号変換装置。

【0115】(付記5) 前記第1の音声符号化方式のフレーム長が第2の音声符号化方式のフレーム長の半分であり、第1の音声符号化方式のフレームが2つのサブフレームを含み、第2の音声符号化方式のフレームが4つのサブフレームを含み、かつ、第1の音声符号化方式はサブフレーム毎にピッチラグ符号を n_0, n_1 ビットで表現し、第2の音声符号化方式はサブフレーム毎にピッチラグ符号を $n_0, (n_1+1), n_0, (n_1+1)$ ビットで表現するとき、前記ピッチラグ変換手段は、第1の音声符号化方式の連続する2つのフレームよりピッチラグ符号が順次 n_0, n_1, n_0, n_1 ビットで表現される4個の連続するサブフレームを作成し、第1、第3サブフレームのピッチラグ符号を第2の音声符号化方式の第1、第3サブフレームのピッチラグ符号とし、第2、第4サブフレームのピッチラグ符号に一定値を加算したピッチラグ符号を第2の音声符号化方式の第2、第4サブフレームのピッチラグ符号とする、ことにより、前記第1のピッチラグ符号を第2のピッチラグ符号に変換する、ことを特徴とする付記2記載の音声符号化装置。

【0116】(付記6) 前記第1の音声符号化方式のフレーム長が第2の音声符号化方式のフレーム長の半分

であり、第1の音声符号化方式のフレームが2つのサブフレームを含み、第2の音声符号化方式のフレームが4つのサブフレームを含み、かつ、第1の音声符号化方式はサブフレーム毎に雑音符号を m_1 , m_1 ビットで表現し、第2の音声符号化方式はサブフレーム毎に雑音符号を m_1 , m_1 , m_1 , m_1 ビットで表現するとき、前記雑音符号変換手段は、第1の音声符号化方式の連続する2つのフレームより雑音符号が順次 m_1 , m_1 , m_1 , m_1 ビットで表現される4個の連続するサブフレームを作成し、これら第1～第4サブフレームの雑音符号を第2の音声符号化方式の第1～第4サブフレームの雑音符号とすることにより、前記第1の雑音符号を第2の雑音符号に変換する、ことを特徴とする付記2記載の音声符号化装置。

【0117】(付記7) 音声信号の一定サンプル数をフレームとし、フレーム毎の線形予測分析により得られる線形予測係数(LPC係数)又は該LPC係数から求まるLSPパラメータを量子化することにより得られる第1のLPC符号と、周期性音源信号を出力するための適応符号帳の出力信号を特定する第1のピッチラグ符号と、雑音性音源信号を出力するための雑音符号帳の出力信号を特定する第1の雑音符号と、前記適応符号帳の出力信号の振幅を表す適応符号帳ゲインを量子化して得られる第1の適応符号帳ゲイン符号と、前記雑音符号帳の出力信号の振幅を表す雑音符号帳ゲインを量子化して得られる第1の雑音符号帳ゲイン符号を求め、これら符号で音声信号を符号化する方式を第1の音声符号化方式とし、第1の音声符号化方式と異なる量子化方法により量子化して得られる第2のLPC符号、第2のピッチラグ符号、第2の雑音符号、第2のゲイン符号とで音声信号を符号化する方式を第2の音声符号化方式とすると、第1の音声符号化方式により符号化した音声符号を入力され、該音声符号を第2の音声符号化方式の音声符号に変換する音声符号変換装置において、前記第1のLPC符号を第1の音声符号化方式のLPC逆量子化方法により逆量子化し、得られるLPC係数の逆量子化値を第2の音声符号化方式のLPC量子化テーブルを用いて量子化して第2のLPC符号を求めるLPC符号変換手段、第1の音声符号化方式におけるピッチラグ符号と第2の音声符号化方式におけるピッチラグ符号との相違を考慮した変換処理により、前記第1のピッチラグ符号を第2のピッチラグ符号に変換するピッチラグ変換手段、第1の音声符号化方式における雑音符号と第2の音声符号化方式における雑音符号との相違を考慮した変換処理により、前記第1の雑音符号を第2の雑音符号に変換する雑音符号変換手段、前記第1の適応符号帳ゲイン符号を第1の音声符号化方式のゲイン逆量子化方法により逆量子化して得られる逆量子化値と、前記第1の雑音符号帳ゲイン符号を第1の音声符号化方式のゲイン逆量子化方法により逆量子化して得られる逆量子化値をまとめて、第2の音声符号化方式のゲイン量

子化テーブルを用いて量子化して前記第2のゲイン符号を発生するゲイン符号変換手段、を有することを特徴とする音声符号変換装置。

【0118】(付記8) 前記LPC符号変換手段は、前記第1のLPC符号の逆量子化値と前記求めた第2のLPC符号の逆量子化値との間の第1の距離を演算する第1演算部、現フレームの第2のLPC符号の逆量子化値と前フレームの第2のLPC符号の逆量子化値を用いて中間の第2のLPC符号の逆量子化値を補間演算する補間部、中間の第1のLPC符号の逆量子化値と前記補間により求まる中間の第2のLPC符号の逆量子化値との間の第2の距離を演算する第2演算部、第1、第2の距離の和が最小となるように、前記LPC係数の逆量子化値を第2のLPC符号に符号化する符号部、を備えたことを特徴とする付記2又は付記7記載の音声符号変換装置。

【0119】(付記9) 第1、第2の距離に重み付けする重み付け手段を備え、前記符号部は、重み付けされた第1、第2の距離の和が最小となるように、前記LPC係数の逆量子化値を第2のLPC符号に符号化する、ことを特徴とする付記8記載の音声符号変換装置。

【0120】(付記10) 前記LPC符号変換手段は、LPC係数又はLSPパラメータを n 次のベクトルで表現し、 n 次のベクトルを複数の小ベクトルに分割した時、小ベクトル毎に前記第1、第2の距離の和が小さい複数の符号候補を算出する符号候補算出手段、各小ベクトルの複数の符号候補の中から1つずつ符号を選んでLPC係数逆量子化値の n 次のLPC符号とすると、前記第1、第2の距離の和が最小となる n 次のLPC符号を決定し、該LPC符号を前記第2の符号とするLPC符号決定手段、を備えたことを特徴とする付記8又は付記9記載の音声符号変換装置。

【0121】(付記11) 前記第1の音声符号化方式のフレーム長が第2の音声符号化方式のフレーム長の2倍であり、第1の音声符号化方式のフレームが4つのサブフレームを含み、第2の音声符号化方式のフレームが2つのサブフレームを含み、かつ、第1の音声符号化方式はサブフレーム毎にピッチラグ符号を

n_0 , (n_1+1) , n_0 , (n_1+1) ビット

で表現し、第2の音声符号化方式はサブフレーム毎にピッチラグ符号を n_0 , n_1 ビットで表現するとき、前記ピッチラグ変換手段は、第1音声符号化方式の4個の連続するサブフレームにおける

n_0 , (n_1+1) , n_0 , (n_1+1) ビット

で表現されるピッチラグ符号のうち、第1、第3サブフレームのピッチラグ符号を第2の音声符号化方式の連続する第1、第2フレームの第1サブフレームのピッチラグ符号とし、第2、第4サブフレームのピッチラグ符号から一定値を減算したピッチラグ符号を第2の音声符号化方式の連続する第1、第2フレームの第2サブフレームのピッチラグ符号とする、ことにより前記第1のピッ

チラグ符号を第2のピッチラグ符号に変換する、ことを特徴とする付記7記載の音声符号化装置。

【0122】(付記12) 前記第1の音声符号化方式のフレーム長が第2の音声符号化方式のフレーム長の2倍であり、第1の音声符号化方式のフレームが4つのサブフレームを含み、第2の音声符号化方式のフレームが2つのサブフレームを含み、かつ、第1の音声符号化方式は4つのサブフレームの各雑音符号を

m_1, m_1, m_1, m_1 ビット

で表現し、第2の音声符号化方式は2つのサブフレームの各雑音符号を m_1, m_1 ビットで表現するとき、前記雑音符号変換手段は、第1の音声符号化方式の第1、第2サブフレームの雑音符号を第2音声符号化方式の第1フレームの第1、第2サブフレームの雑音符号とし、第1の音声符号化方式の第3、第4サブフレームの雑音符号を第2音声符号化方式の第2フレームの第1、第2サブフレームの雑音符号とすることにより、前記第1の雑音符号を第2の雑音符号に変換する、ことを特徴とする付記7記載の音声符号化装置。

【0123】(付記13) フレーム毎に音響信号を第1の符号化方式により符号化して得られる音響符号を入力され、該音響符号を第2の符号化方式の音響符号に変換して出力する音響符号変換装置において、第1の音響符号化方式による音響符号より、音響信号を再現するために必要な複数の成分の符号を分離する符号分離手段、分離された複数の成分の各符号を第2の音響符号化方式の音響符号に変換する符号変換部、伝送路誤りが発生していなければ分離された符号をそのまま符号変換部に入力し、伝送路誤りが発生していれば誤り隠蔽処理を施して得られる符号を符号変換部に入力する符号修正部各符号変換部から出力する符号を多重して第2の音響符号化方式による音響符号を出力する手段、を備えたことを特徴とする音響符号変換装置。

【0124】(付記14) 音響信号の一定サンプル数をフレームとし、フレーム毎の線形予測分析により得られる線形予測係数(LPC係数)又は該LPC係数から求まるLSPパラメータを量子化することにより得られる第1のLPC符号と、周期性音源信号を出力するための適応符号帳の出力信号を特定する第1のピッチラグ符号と、雑音性音源信号を出力するための代数符号帳の出力信号を特定する第1の代数符号と、前記適応符号帳の出力信号の振幅を表すピッチゲインと前記代数符号帳の出力信号の振幅を表す代数符号帳ゲインとを量子化して得られる第1のゲイン符号とを求め、これら符号で音響信号を符号化する方式を第1の音響符号化方式とし、第1の音響符号化方式と異なる量子化方法により量子化して得られる第2のLPC符号、第2のピッチラグ符号、第2の代数符号、第2のゲイン符号とで音響信号を符号化する方式を第2の音響符号化方式とすると、第1の音響符号化方式により符号化した音響符号を入力され、該音響符号を第2

の音響符号化方式の音響符号に変換する音響符号変換装置において、第1の音響符号化方式による音響符号より、音響信号を再現するために必要な複数の成分の符号を分離する符号分離手段、分離された複数の成分の各符号を第2の音響符号化方式の音響符号に変換する符号変換部、伝送路誤りが発生していなければ分離された符号をそのまま符号変換部に入力し、伝送路誤りが発生していれば誤り隠蔽処理を施して得られる符号を符号変換部に入力する符号修正部各符号変換部から出力する符号を多重して第2の音響符号化方式による音響符号を出力する手段、を備えたことを特徴とする音響符号変換装置。

【0125】(付記15) 前記符号修正部は、現フレームに伝送路誤りが発生した場合、過去のLPC逆量子化値で現フレームのLPC逆量子化値を推定し、前記符号変換部は該推定されたLPC逆量子化値から第2の音響符号化方式の現フレームにおけるLPC符号を求める、ことを特徴とする付記14記載の音響符号変換装置。

(付記16) 前記符号修正部は、現フレームに伝送路誤りが発生した場合、過去のピッチラグ符号を現フレームのピッチラグ符号とすることで前記誤り隠蔽処理を実行し、前記符号変換部は該過去のピッチラグ符号から第2の音響符号化方式の現フレームにおけるピッチラグ符号を求める、ことを特徴とする付記14記載の音響符号変換装置。

(付記17) 前記符号修正部は、現フレームに伝送路誤りが発生した場合、過去の代数符号を現フレームの代数符号とすることで前記誤り隠蔽処理を実行し、前記符号変換部は該過去の代数符号から第2の音響符号化方式の現フレームにおける代数符号を求める、ことを特徴とする付記14記載の音響符号変換装置。

【0126】(付記18) 前記符号修正部は、現フレームに伝送路誤りが発生した場合、過去のゲイン符号から現フレームのゲイン符号を推定し、前記符号変換部は該推定されたゲイン符号から第2の音響符号化方式の現フレームにおけるゲイン符号を求める、ことを特徴とする付記14記載の音響符号変換装置。

(付記19) 前記符号修正部は、現フレームに伝送路誤りが発生した場合、過去のピッチゲイン逆量子化値よりピッチゲイン G_a を求めると共に、過去の代数符号帳ゲイン逆量子化値より代数符号帳ゲイン G_c を求め、前記符号変換部はこれらピッチゲイン G_a と代数符号帳ゲイン G_c とから第2の音響符号化方式の現フレームにおけるゲイン符号を求める、ことを特徴とする付記14記載の音響符号変換装置。

【0127】

【発明の効果】以上本発明によれば、第1の符号化方式による音声符号より音声信号を再現するために必要な複数の成分の符号を分離し、各成分の符号をそれぞれ逆量子化し、逆量子化値を第2の符号化方式により量子化して符号変換するため、音声符号を音声に復号することな

しに音声符号の変換ができる。このため、従来のタンデム接続に比べて遅延時間を小さくすることができ、しかも、音質劣化を小さくすることができる。

【0128】また、本発明によれば、LSP符号の変換に際して、第1符号化方式のLSP符号を逆量子化し、逆量子化値LSP₁(i)を第2符号化方式の量子化テーブルを用いて量子化して符号変換する場合、該逆量子化値LSP₁(i)と変換により得られたLSP符号の逆量子化値LSP_c3(i)間の第1の距離(誤差)だけでなく、中間の第1符号化方式のLSP符号逆量子化値LSP₀(i)と補間により演算した中間の第2符号化方式のLSP符号逆量子化値LSP_c1(i)間の第2の距離(誤差)をも考慮に入れてLSP符号変換するため、フレーム内で入力音声の性質が変化するような場合でも変換誤差の少ない良好な音声符号変換を行うことができる。又、本発明によれば、第1、第2の距離に重み付けし、重み付けされた第1、第2の距離の和が最小となるように、LPC係数逆量子化値LSP₁(i)を第2の符号化方式におけるLPC符号に符号化するため、より変換誤差の少ない良好な音声符号変換を行うことができる。

【0129】又、本発明によれば、LPC係数又はLSPパラメータをn次のベクトルで表現すると共に、n次のベクトルを複数の小ベクトル(低域、中域、高域小ベクトル)に分割し、小ベクトル毎に前記第1、第2の距離の和が小さい複数の符号候補を算出し、各小ベクトルの複数の符号候補の中から1つずつ符号を選んでn次のLPC符号とし、前記第1、第2の距離の和が最小となる組み合わせに基づいてn次のLPC符号を決定するため、より高い音質の再生が可能な音声符号変換ができる。また、本発明によれば従来の音声符号変換部で問題となっていた回線誤りによる音質劣化を減少させ、変換後の良好な再生音声を提供することができる。特に、近年の低ビットレート音声符号化で広く用いられているCELPアルゴリズムでは、音声合成フィルタとして用いるIIRフィルタを用いるため、回線誤りの影響を受けやすく、しばしば発振現象により大きな異音を発生するため、本発明による改善効果は非常に大きい。

【図面の簡単な説明】

【図1】本発明の第1の原理図である。

【図2】本発明の第2の原理図である。

【図3】第1実施例の構成図である。

【図4】LSP量子化されるフレームの説明図である。

【図5】LSP量子化器の構成図である。

【図6】フレームとサブフレームの対応図である。

【図7】G.729A方式のピッチラグとインデックスの関係図である。

【図8】AMR方式のピッチラグとインデックスの関係図である。

【図9】G.729A方式のピッチラグとAMR方式のピッチラグの対応関係図である。

【図10】ゲイン量子化器の構成図である。

【図11】第2実施例の構成図である。

【図12】第2実施例のLSP量子化器の構成図である。

【図13】第2実施例のLSP符号化処理フローである。

【図14】第3実施例の処理フロー(その1)である。

【図15】第3実施例の処理フロー(その2)である。

【図16】第4実施例の構成図である。

【図17】第4実施例のLSP符号変換部の処理説明図である。

【図18】LSP逆量子化器の構成図である。

【図19】LSP量子化器の構成図である。

【図20】音声符号に回線誤りが混入する場合の説明図である。

【図21】第5実施例の原理説明図である。

【図22】第5実施例の構成図である。

【図23】ITU-T G.729A符号変換方式による符号器の構成図である。

【図24】量子化方法説明図である。

【図25】適応符号帳の説明図である。

【図26】G.729A符号変換方式における代数符号帳の説明図である。

【図27】各パルス系統グループのサンプリング点説明図である。

【図28】ITU-T G.729A符号変換方式による復号器の構成図である。

【図29】ITU-T G.729A符号変換方式とAMR符号変換方式における主要緒元の比較図である。

【図30】フレーム長比較図である。

【図31】ITU-T G.729A符号変換方式とAMR符号変換方式におけるビット割り当ての比較図である。

【図32】従来技術の概念図である。

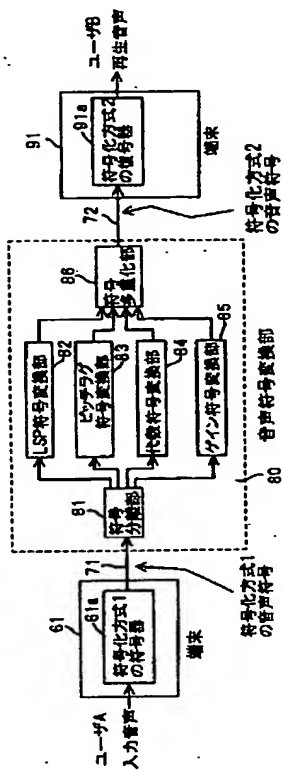
【図33】従来技術の音声符号変換例である。

【符号の説明】

- 61・・・端末
- 61a・・・符号化方式1の符号器
- 71, 72・・・伝送路
- 80・・・音声符号変換部
- 81・・・符号分離部
- 82・・・LSP符号変換部
- 83・・・ピッチラグ符号変換部
- 84・・・代数符号変換部
- 85・・・ゲイン符号変換部
- 86・・・符号多重化部
- 91・・・端末
- 91a・・・符号化方式2の復号器

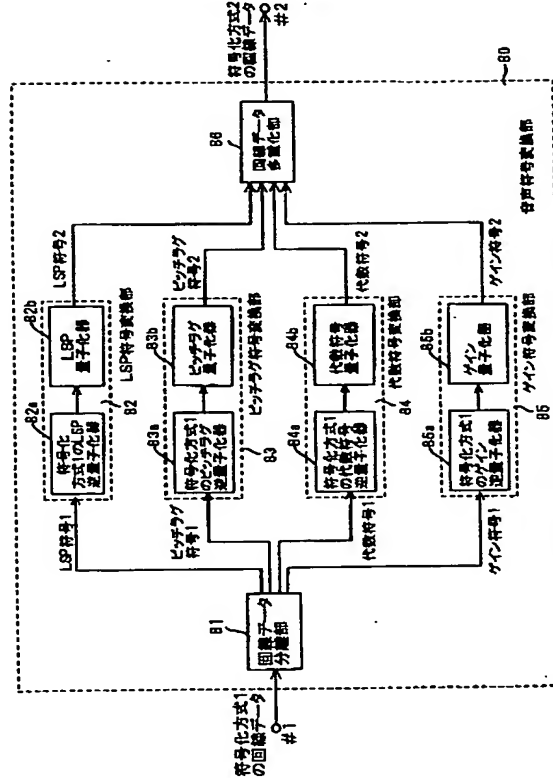
【図1】

本発明の第1の原理図



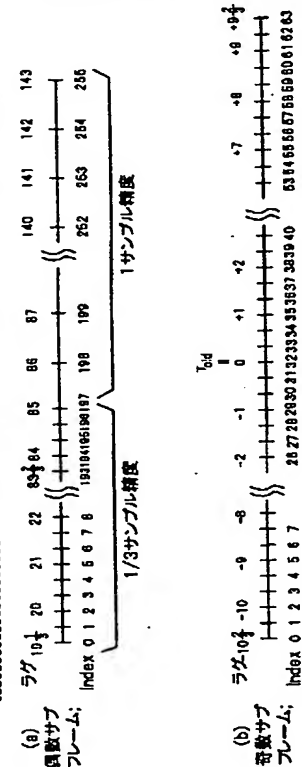
【図2】

本発明の第2の原理図



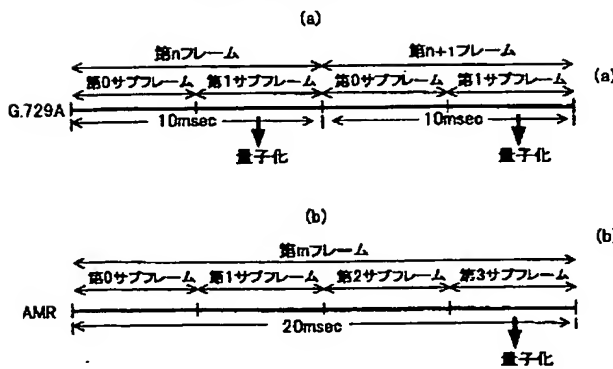
【図8】

AMRのピッチラグとインデックスの関係



【図4】

LSP量子化とフレームの関係



【図6】

フレームとサブフレームの対応

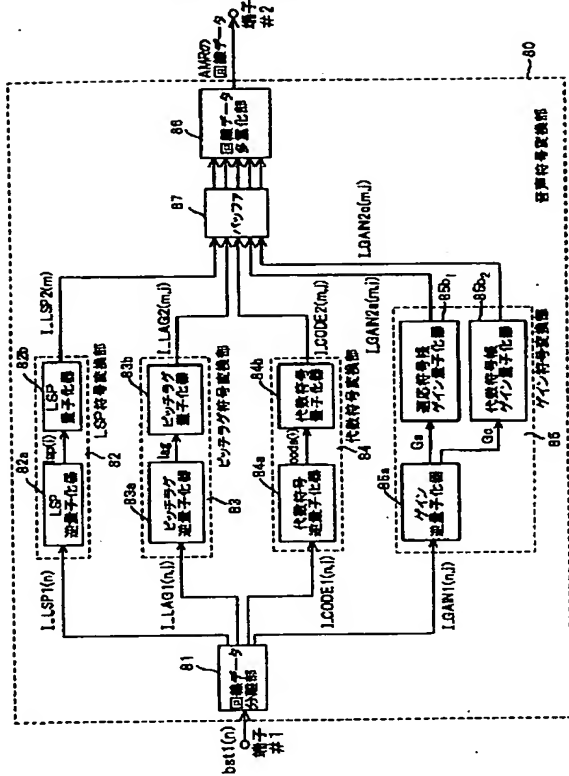
サブフレーム番号/フレーム番号				
G.729	0/n	1/n	0/(n+1)	1/(n+1)
AMR	0/m	1/m	2/m	3/m

各サブフレームにおけるピッチラグ符号のビット数の対応				
ピッチラグ符号(ビット数)				
G.729	8	5	8	5
AMR	8	6	8	6

各サブフレームにおける代数符号のビット数の対応				
代数符号				
G.729	17	17	17	17
AMR	17	17	17	17

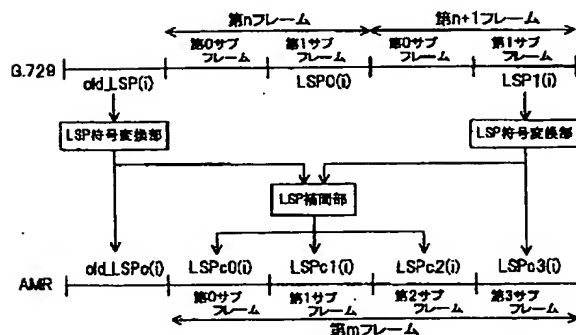
【図3】

実施例1の構成例図



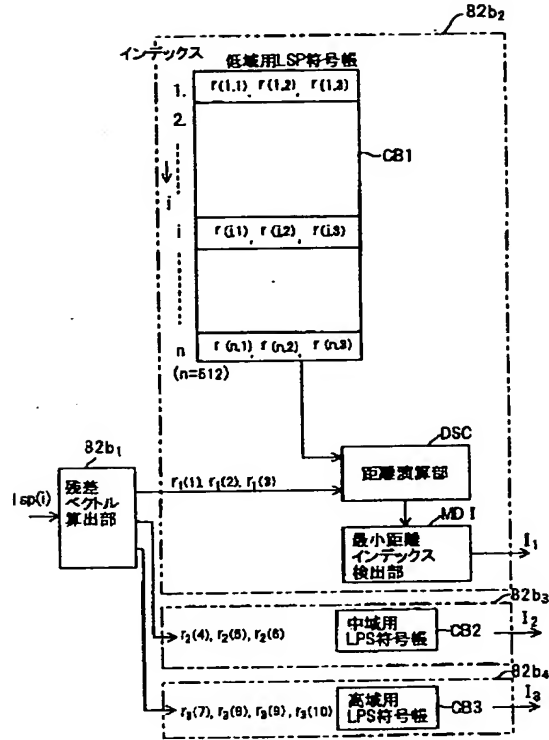
【図11】

実施例2の構成例図



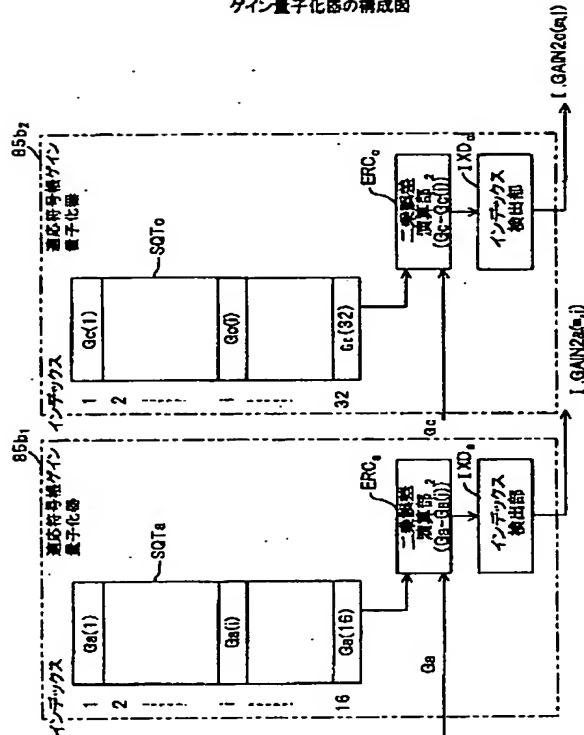
【図5】

LSP量子化器の構成



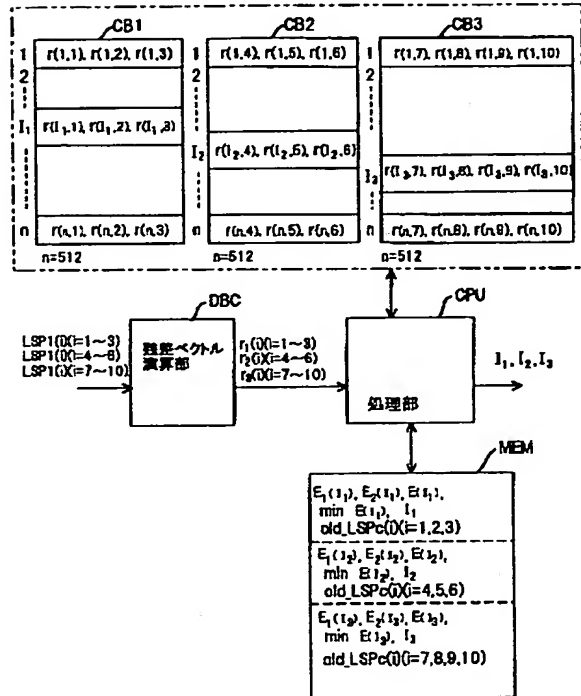
【図10】

ゲイン量子化器の構成図



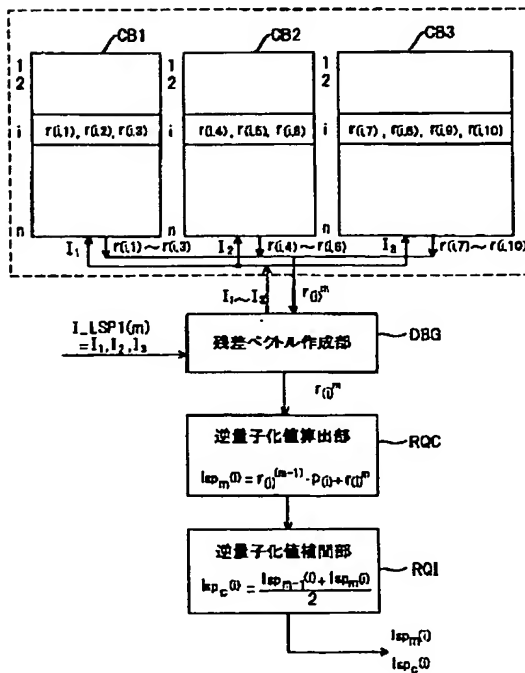
【図12】

第2実施例のLSP量子化器の構成図



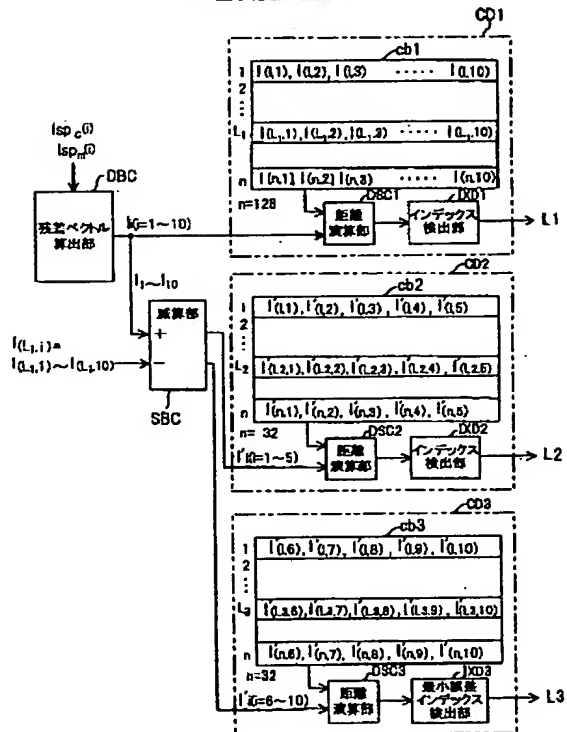
【図18】

LSP逆量子化器の構成



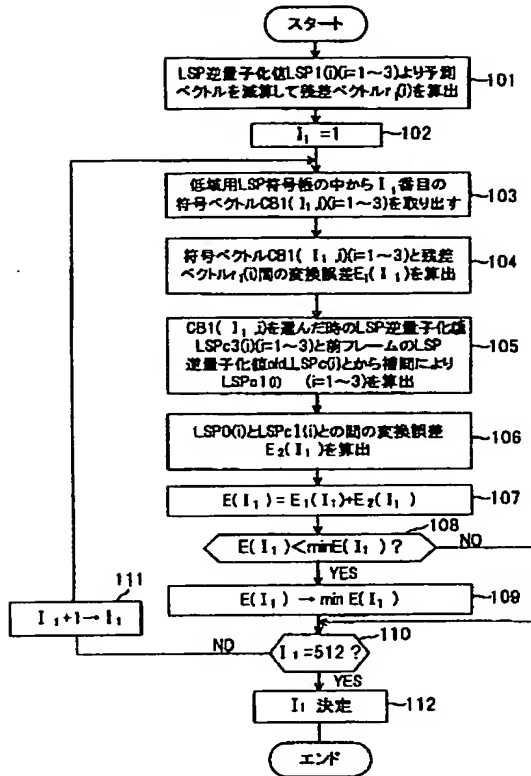
【図19】

LSP量子化器の構成



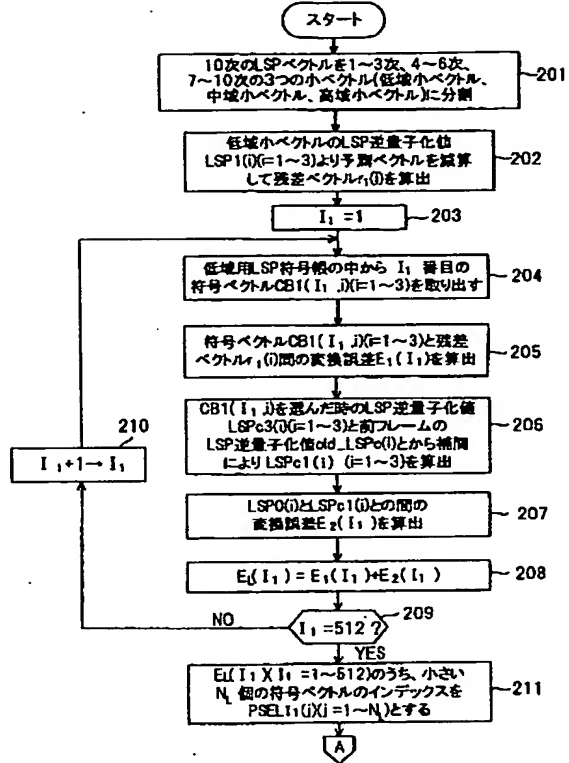
【図13】

第2実施例のLSP符号化処理フロー



【図14】

第3実施例処理フロー(その1)



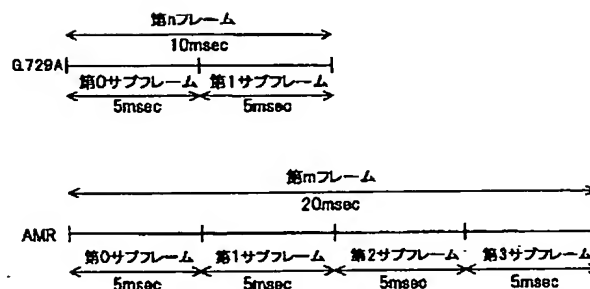
【図29】

ITU-T G.729AとGSM-AMRの比較1(主要諸元)

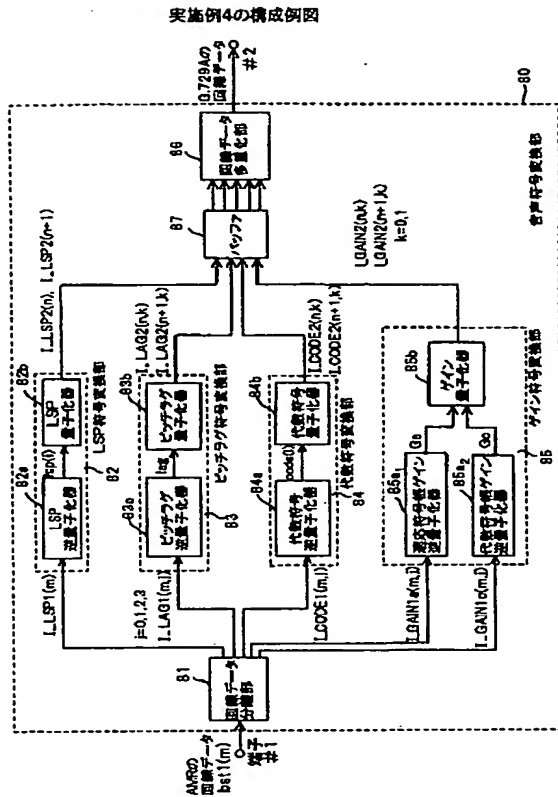
	ITU-T G.729A	GSM-AMR
標本化周波数	8kHz	8kHz
フレーム長	10msec	20msec
サブフレーム長	5msec	5msec
サブフレーム数	2	4
原理遅延	15msec	20msec
線形予測次数	10次	10次

【図30】

フレーム長の比較

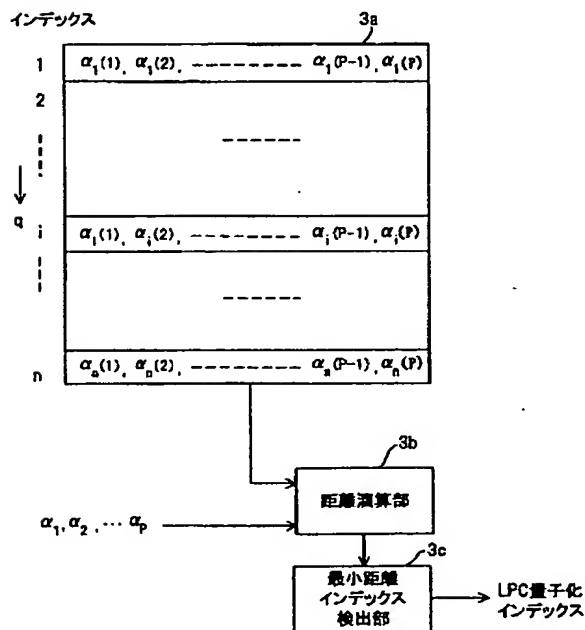


【図16】



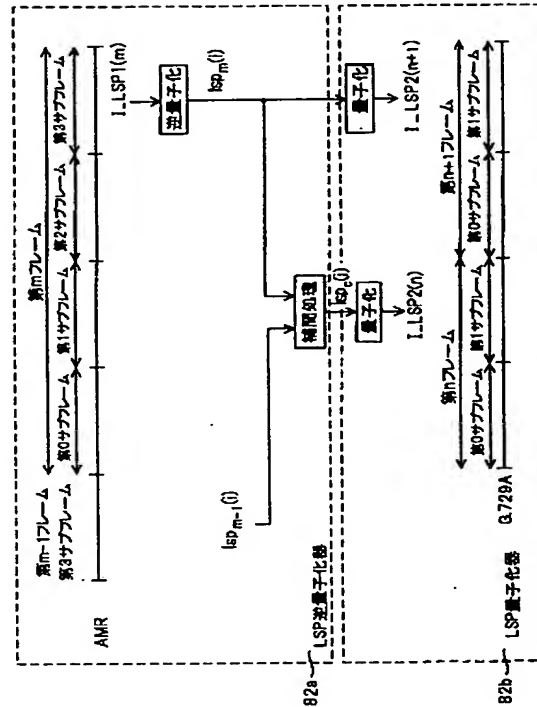
【図24】

量子化方法説明図



【図17】

実施例4のLSP符号変換部の処理



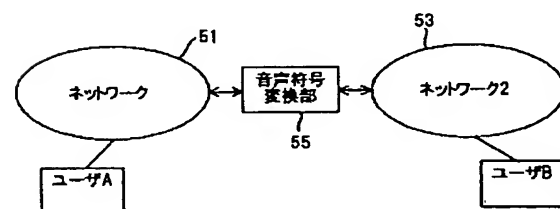
【図31】

ITU-T G.729AとGSM-AMRの比較2(ビット割り当て)

パラメータ	ITU-T G.729A (サブフレーム/フレーム)	AMR(7.95kb/sモード) (サブフレーム/フレーム)
LSP符号	-/18	-/27
ピッチラグ符号	8+5/13	8+6+8+6/28
ピッチパラリティ	1/1	—
代数符号	17+17/34	17+17+17+17/68
ゲイン符号	7+7/14	—
連応符号係数ゲイン符号	—	4+4+4+4/16
代数符号係数ゲイン符号	—	5+5+5+5/20
合計	80bit/10msec	158bit/20msec

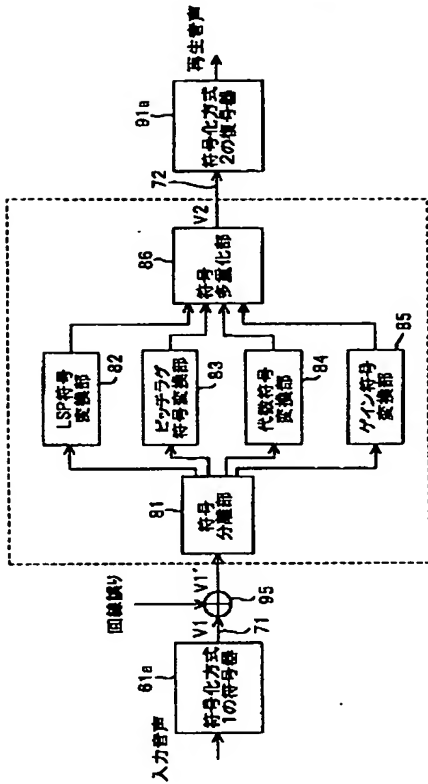
【図32】

従来技術の概念図



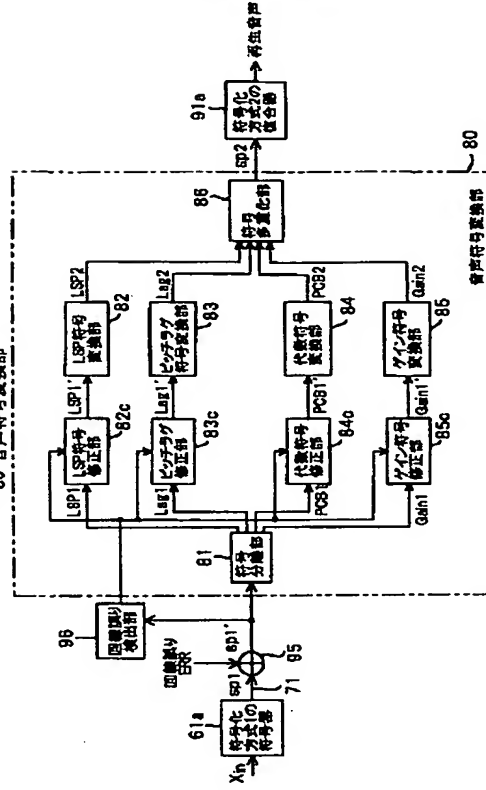
【図20】

回線誤りの混入説明図



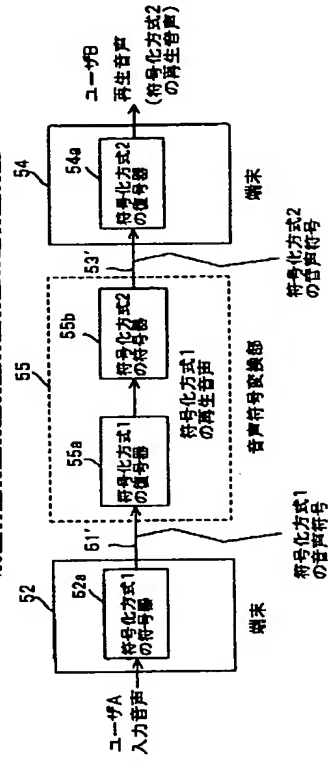
【図21】

第5実施例の原理図



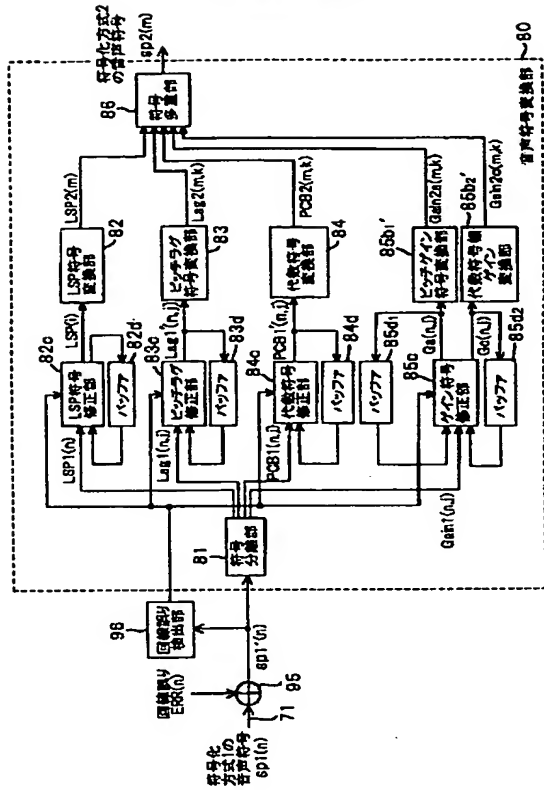
【図33】

従来技術の例



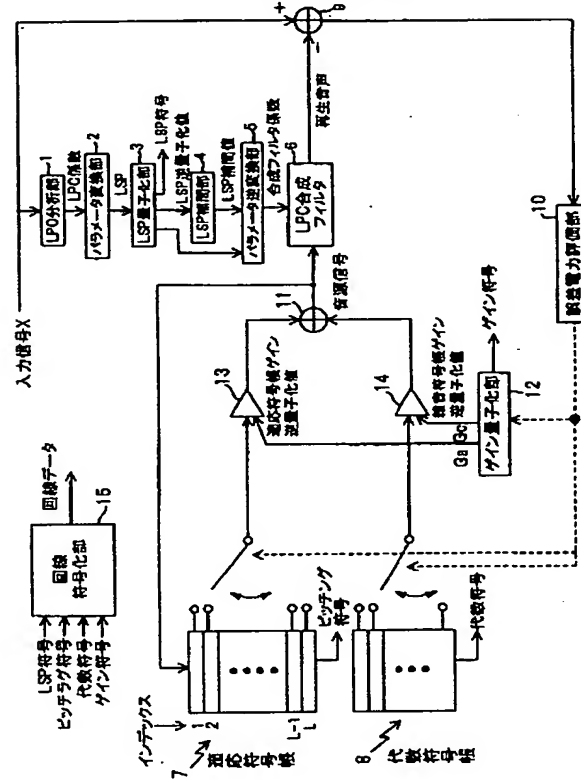
【图22】

第5実施例の構成図



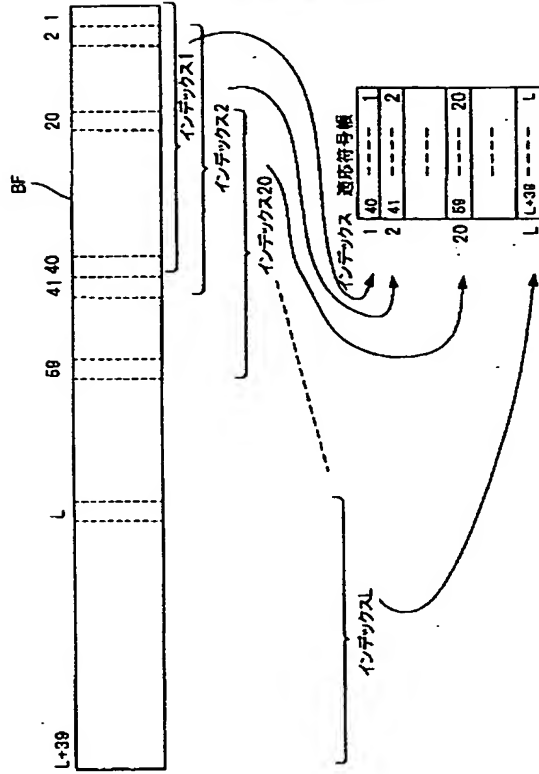
【圖23】

ITU-T G.729Aの符号器の構成



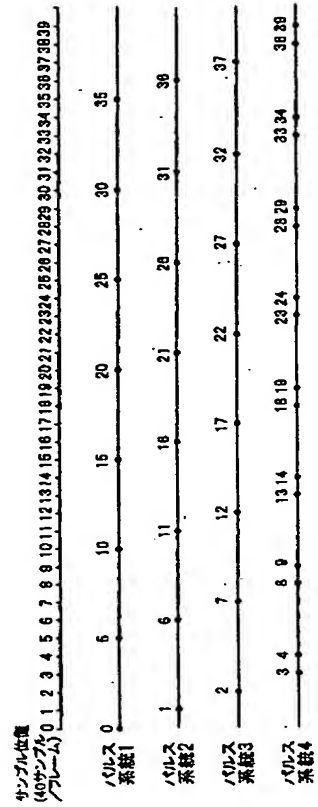
【図25】

適応符号帳の説明図



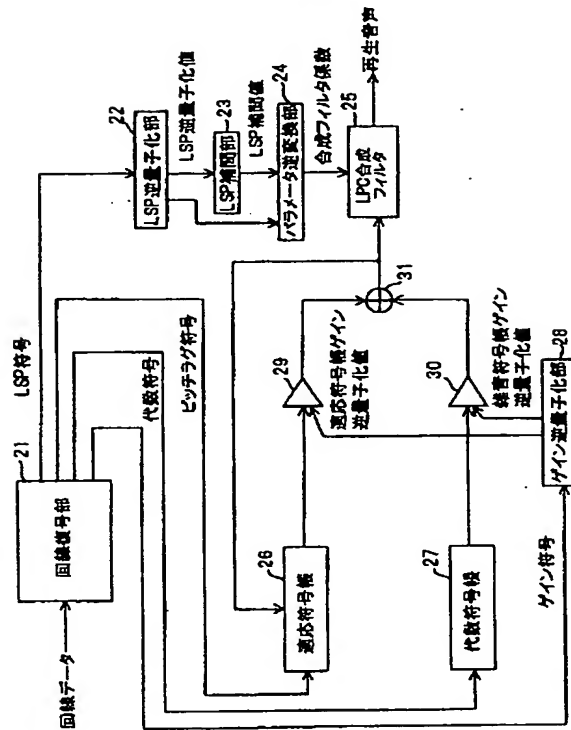
【図27】

各パルス系統グループのサンプリング点説明図



【図28】

ITU-T G.729Aの復号器の構成



フロントページの続き

(72)発明者 土永 義照
 福岡県福岡市博多区博多駅前三丁目22番8
 号 富士通九州デジタル・テクノロジー株
 式会社内

Fターム(参考) 5D045 CA01 CC02 DA06 DA11